

REPORT DOCUMENTATION PAGE			Form Approved OMB NO. 0704-0188	
Public Reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comment regarding this burden estimates or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE 16 March 2004		3. REPORT TYPE AND DATES COVERED Final Technical Report 27-November-2000 -> 26 June 2003
4. TITLE AND SUBTITLE Sensor Modeling and 3D Visualization			5. FUNDING NUMBERS Contract DAAD 19-01-1-0002	
6. AUTHOR(S) James Bethel and Joseph Spann				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Purdue University, School of Civil Engineering, West Lafayette, IN and BAE Systems - Mission Solutions, San Diego, CA (subcontractor)			8. PERFORMING ORGANIZATION REPORT NUMBER ARO-004	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U. S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211			10. SPONSORING / MONITORING AGENCY REPORT NUMBER 41876.1 - EV	
11. SUPPLEMENTARY NOTES The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.				
12 a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12 b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Two related, but independent investigations were carried out under this project. The first related to sensor (camera) modeling as needed for the photogrammetric exploitation of imagery for mapping and geopositioning purposes. This task was further subdivided into two subtasks. The first of these involved performing a geometric camera calibration for a set of 4 co-boresighted video cameras (called CAMIS) used by TEC (Topographic Engineering Center) for remote sensing and mapping purposes. This involved setting up a laboratory calibration facility, collecting system imagery in that facility, developing software and algorithms to align the fixtures in the facility, and developing software and algorithms to perform the camera calibration. The second subtask was to investigate the detection of errors in aerial triangulation, motivated by recurrent problems using the SOCET SET triangulation module for large blocks of images. The approach here was to implement and test two algorithms for such error detection in a prototype MATLAB GUI (graphical user interface). Testing showed that both algorithms had merit for the error/blunder detection problem. The second task in the project was the further development of Vesper for the Battlespace (VFB) product. The goals here were to develop and demonstrate advanced sensor data fusion, 2D and 3D visualization capabilities, combine the visualization environment with precise geo-registration, creating a single, end-to-end data visualization system. This system enhances time critical operations such as intelligence gathering, mission planning and rehearsal, and operational command and control. Specific modules within VFB include auto geo-registration of EO,SAR,IR,HSI, and SIG data, sensor model update, sensor projectors, object models (DTED, etc.), and interactive 2D/3D visualization.				
14. SUBJECT TERMS Sensor model, camera, calibration, CAMIS, aerial triangulation, blunder detection, robust estimation, visualization, registration, fusion, graphics			15. NUMBER OF PAGES 112	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OR REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION ON THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

SENSOR MODELING AND 3D VISUALIZATION FINAL TECHNICAL REPORT

Foreward

Note that this actual report is very brief, with the bulk of the technical material located in previously produced papers, reports, and presentations, that are attached as appendices.

Appendices

- A. Paper describing the CAMIS calibration project.
- B. Technical Report describing the CAMIS calibration project
- C. Presentation slides giving and overview of the CAMIS calibration project.
- D. Report describing the work done for the aerial triangulation blunder detection project
- E. Presentation slides for VESPER, VFB (1)
- F. Presentation slides for VESPER, VFB (2)
- G. Presentation slides for VESPER, VFB (3)

Statement of the Problem Studied

There were several related but independent problems that were studied for this project. The three principal ones were: (1) Determine a practical laboratory calibration procedure for small format aerial camera systems such as the CAMIS system, (2) Determine a practical algorithm for blunder detection in the processing of aerial photogrammetric blocks, and (3) Develop and enhance a visualization environment where new imagery can be imported, registered, and viewed in a 3D multi-modal setting.

Publications (peer reviewed conference proceedings)

- 1. Alharthy, A., and Bethel, J., 2002, "Geometric Calibration of the CAMIS Sensor", presented at the FIG/ACSM/ASPRS Conference, April 25, 2002, Washington DC
- 2. Alharthy, A. and Bethel, J., 2002, "Laboratory Calibration of a Multiband Sensor", presented at ISPRS Commission III conference, Sept. 2002, Graz, Austria

Personnel

- 1. James Bethel co-PI
- 2. Joseph Spann co-PI
- 3. Abdullatif Alharthy, PhD received, August 2003
- 4. Ade Mulyana, PhD expected, August 2004
- 5. Junhee Youn, PhD expected, May 2005
- 6. Karen Kaufman

Inventions

None. (See earlier form DOD822)

Bibliography

See individual papers and reports for bibliography sections.

GEOMETRIC CALIBRATION OF THE CAMIS SENSOR

Abdullatif Alharthy, James Bethel
School of Civil Engineering, Purdue University
1284 Civil Engineering Building
West Lafayette, IN 47907
alharthy@ecn.purdue.edu
bethel@ecn.purdue.edu

ABSTRACT:

CAMIS is a system for airborne remote sensing and is designed to utilize modern solid-state imaging and data acquisition technology. It is composed of four CCD cameras with band pass optical filters to obtain four band images. In this paper, we summarize the work that has been done during the geometric calibration of the CAMIS sensor. We modified the conventional calibration procedure especially for this sensor and we modified and used a matching technique to make the process more efficient. A network bundle adjustment program was developed and used to adjust the laboratory measurements and locate the targets. Images of the target field were then taken by each of the four cameras of the CAMIS sensor. Two matching techniques were used to determine and refine the target locations in the image space. We modified the matching algorithm to overcome certain radiometric effects and thereby found the location of the target centers in image space.

To recover the most significant camera parameters, a full math model was used. The unified least squares approach was used iteratively to solve this nonlinear overdetermined system. In order to determine the lens distortion behavior, the radial and decentering components were estimated. Then the radial distortion curve was equalized and the corresponding changes to the sensor parameters were recorded. Finally, we present four sets of adjusted parameters, one per camera.

1. INTRODUCTION

The CAMIS sensor consists of four co-boresighted area-CCD cameras with band pass filters: blue, green, red, and near infrared. In this paper, we summarize the work that has been done during the geometric calibration of the CAMIS sensor. The procedure required many preliminary steps such as preparing the calibration site which involved target layout, setting up the coordinate system and locating fiducial monuments within that system. Three arc-second theodolites and a steel tape were used to measure the angles and distances in the network of calibration targets. In order to adjust those measurements and to get the target coordinates into the reference coordinate system, we developed a network bundle adjustment program. Images of the target field were then taken by each of the four cameras of the CAMIS sensor. The coordinates of the targets in both the object and the image system were used as observations for estimating the sensor parameters in a second bundle program configured for self-calibration.

After planning the data flow, the images were taken and the calibration procedure was started. To cover the most significant parameters, a full math model was used (Samtaney, 1999). A description of this model and its use are outlined in this paper. The unified least square approach was used iteratively to solve this nonlinear overdetermined system since we have some prior knowledge about a number of the sensor parameters (Mikhail and Ackerman, 1976). Moreover, in order to see the distortion behavior, the radial and decentering distortions were calculated and plotted separately. Afterward, the radial distortion curve was equalized and the corresponding changes to the sensor parameters were recorded. We repeated the procedure for each camera individually and consequently our results have four sets of adjusted parameters, one per camera. The basic steps and algorithms that were used during the calibration process are outlined below. In the actual use of this imaging system, often three of the bands are registered and resampled to a reference band. In that case, only the calibration of that reference band would be used.

2. CALIBRATION

The aim of this work was to make a laboratory calibration for the geometric parameters of the CAMIS sensor. CAMIS stands for Computerized Airborne Multicamera Imaging System. It is designed to utilize modern solid-state imaging and data acquisition technology. It is composed of four CCD cameras with band pass optical filters to obtain four band images. The center wavelength of those bands is as follows: 450, 550, 650 and 800 nm. However, each sensor has its own optics and obtains its own image independently from the others at the same time. Those four images can be integrated into a composite image or viewed individually. The CAMIS sensor has been used in multispectral imaging and mapping purposes by mounting it in an airplane with GPS and INS systems. These auxiliary sensors provide very good position and attitude data for stabilizing the subsequent bundle block adjustment. The calibration procedure required a number of steps and they are summarized below.

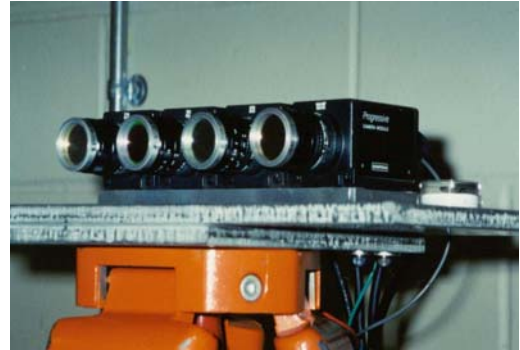


Figure 1: the four cameras

2.1 SITE PREPARATION

The idea of the calibration was to layout some targets in the object space, locate them accurately, and acquire an image of those targets by the sensor. Then, we relate the coordinates of the targets in both systems, image and object space, in order to obtain the camera parameters. So, the procedure starts by setting up the calibration site. First, we designed the targets to be cross shapes so their center positions will be obtained very easily. Then they have been laid out in an “X” pattern that allows us to recover the needed geometric parameters and systematic errors as shown in figure 3. Those targets were placed on an almost flat service wall and the sensor position was located around 8 meters away from that wall. In order to register the objects in the scene, an object space coordinate system was established at the site and two other instrument stations were marked to use in the measurements.



Figure 2: sensor setup for calibration

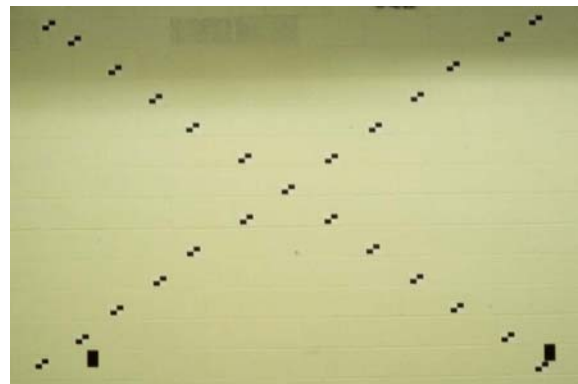


Figure 3: target layout

2.2 MEASUREMENTS AND ADJUSTMENT IN OBJECT SPACE

Two three arc-second theodolites were mounted at the referenced stations and were used to measure directions to all relevant objects of the network: target centers, theodolite locations, and camera case monuments. The origin of the object space was chosen to be theodolite one at station one at the right of the sensor position. Many manual measurements were made of the camera physical layout, using machinist calipers. The lenses were also placed on an

optical bench for determining the locations of the nodal points. A cross section of one camera is shown in figure 4. This was needed to locate the camera front nodal point with respect to the camera body, which would be located in the network by theodolite observations. The spacings between the wall targets were measured with a steel tape. Having all these observations, we end up with an overdetermined system of equations. We developed a bundle program to simultaneously adjust the theodolite and distance observations. As a result of that, we determined all of our targets and camera stations in the referenced coordinate system. The next step was capturing images by the sensor(s).

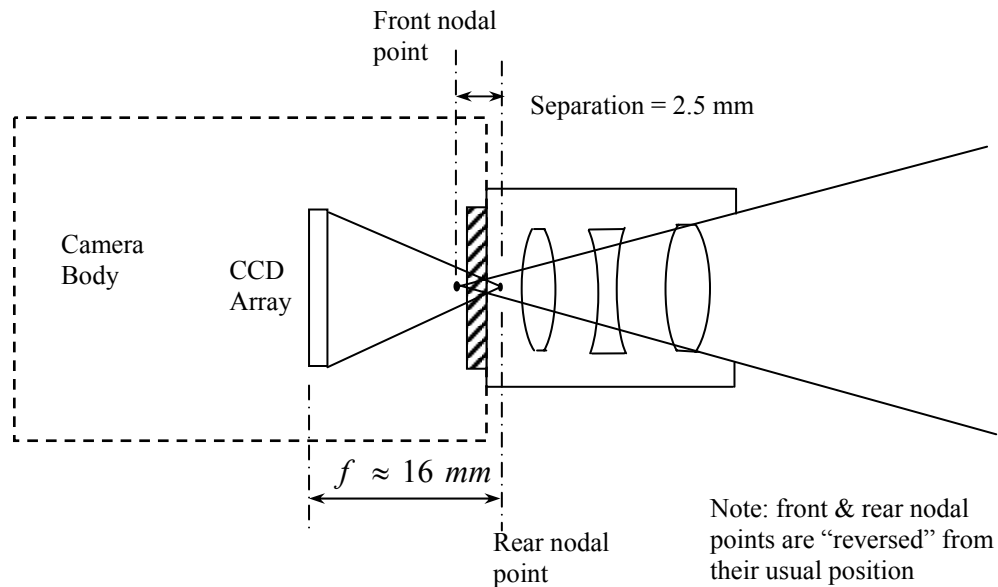


Figure 4: cross section of one camera showing the lenses, rear and front nodal points.

2.3 CAPTURING IMAGES AND OBTAINING IMAGE SPACE COORDINATES

Before the measurements, the sensor was mounted on a leveled plate fixed on a survey tripod. In this sense, the exposure stations were fixed and predetermined to an accuracy of a few millimeters. In this step we tried to simulate the real working conditions by setting the lenses to the “working” infinity focus position. Images were viewed after captured to verify acceptable radiometry. With the band pass filters our illumination setup was just sufficient to produce acceptable image definition for the targets. In the future, stronger light sources would be used to allow more flexibility. After this step, the laboratory work ended and the processing procedure started.

2.4 IMAGE SPACE CALCULATIONS

Once the images were captured, we ran a cross correlation matching program to get rough approximation of target positions in the image space to within a pixel. The cross correlation matching function works by computing the similarity between two same sized windows (Mikhail, Bethel and McGlone, 2001; Mikhail and Ackerman, 1976). One window patch contains the ideal target and the other contains a window from the image. In general, a matching problem is a key algorithm for other applications and image analysis. Despite the fact that, the cross correlation matching results showed that we are only away from the exact position by a pixel or less, we needed more accurate and precise methods to guarantee the sub-pixel precision. This level of precision is necessary for a camera calibration problem. Least squares matching (LSQM) is very adequate technique for this purpose. LSQM utilizes the

first derivative (gradient) of the intensity in both x and y directions to obtain the best correspondence and the exact matching can be reached by moving one window with respect to the other one (Atkinson, 1996). Some obstacles, such as radiometric effects, were faced and solved by modifying the algorithm. As a result of this step, the image space coordinates for the targets have been obtained. Results using both techniques are shown in figure 5.



Figure 5: Target center determination using two matching algorithms.

2.5 CAMERA PARAMETER ESTIMATION

First, a full math model was chosen to perform the geometric calibration and estimate the sensor parameters. In this math model, Ravi Samtaney (1999) tried to cover the significant factors that might occur in geometric calibration. This model is explored more below. The model relates the target coordinates in object space and image space through the camera parameters. This overdetermined and nonlinear system needs an optimization criterion to be solved. Since we have some initial values for a number of the camera parameters and their uncertainty, we decided to use the unified least squares algorithm to solve the system. Using the resulting distortion parameters, plots were drawn to describe the radial and decentering distortion behavior. Finally, the radial distortion curve was equalized by small changes in corresponding parameters. The results of the process for each sensor are tabulated in the results section.

3. TARGET LOCATIONS IN IMAGE SPACE (MATCHING)

3.1 INTRODUCTION

In order to find the target locations in the image, two matching approaches were used. First, the approximate locations are obtained using the cross correlation matching. Second, we refine the results of the first algorithm using least squares matching.

3.2 CROSS-CORRELATION MATCHING

Cross-correlation determines the similarity between two corresponding patches. The conventional cross-correlation approach cannot give the precise location of an object due to many factors. Differences in attitude, distortion, and signal noise are some examples that affect the correspondence in geometry and radiometry (Mikhail, Bethel and McGlone, 2001; Atkinson 1996). However, this algorithm usually gives an approximate location of the correspondence within a few pixels.

The ideal template will be passed through the image and the matching function will be computed and recorded at the center pixel of the patch. The match function, the normalized cross correlation coefficient, ranges between +1 and -1. The maximum value equals +1, which means there is a full match between the two windows or, in other words, they are identical. Usually a threshold will be used to distinguish between matches and non-matches. Then the pixels with a cross correlation above the threshold will be considered candidates for the match. Some individual correlation results, as shown in figure 3, are off from the center of the target only by a pixel or less. These results enable us to use the least squares refining technique directly.

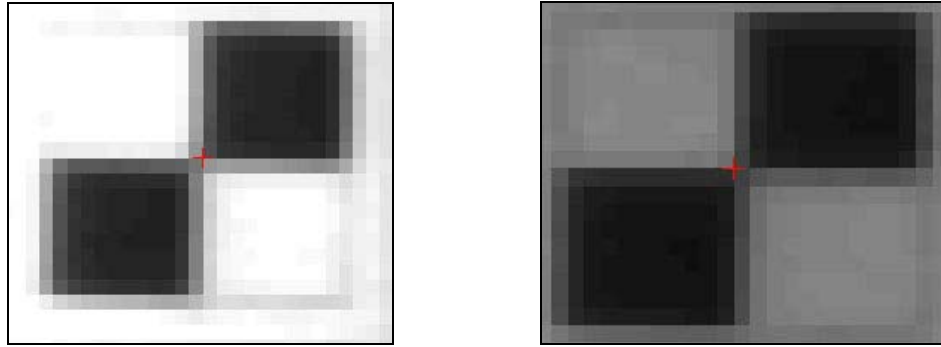


Figure 3: two cross correlation match results.

3.3 LEAST SQUARES MATCHING (LSQM)

Least squares matching works as a powerful and an accurate technique to refine an object's coordinates in the image space based on the correspondence between a image chip and a reference or template chip (Mikhail, Moffitt and Francis, 1980; Atkinson, 1996). This technique utilizes the gradient in the x and y directions in order to move the two patches with respect to each other in order to get the best match. The match precision that we are looking for with this technique is within a hundredth of a pixel or so. The similarity between the two targets was only geometrically modeled for this specific problem since the radiometric differences were eliminated through some preprocessing steps, as we will see below. The problem is to match an ideal shape of the target with a small window from the image containing the imaged target. The following steps describe the automated procedure that was used for setting up the two windows for matching:

- 1) Obtain the approximate location of the imaged target using the first matching approach (cross correlation) as described above. Those locations should be within a few pixels of the exact location in order to make the geometric model in LSQM converge and to produce accurate results.
- 2) Having the rough estimated location, a window around that location from the image with adequate size will be extracted for matching purposes. This is all done systematically inside the code.
- 3) The ideal or template target is retrieved at this point. Similarity in the intensity is enforced between the two windows.

After specifying the two windows with the same size for matching, the LSQM procedure takes place. Requiring similarity in intensity between each of the two corresponding pixels from the two windows is the basic condition for this procedure. Since the two patches do not have the same coordinate system, a 6-parameter geometric transformation is used to relate them in the matching procedure (Atkinson, 1996). Those parameters will be corrected iteratively and will be used to calculate the new coordinates x', y' in order to use them in resampling the grid for the template window. We used bilinear interpolation to resample the intensity values during this procedure.

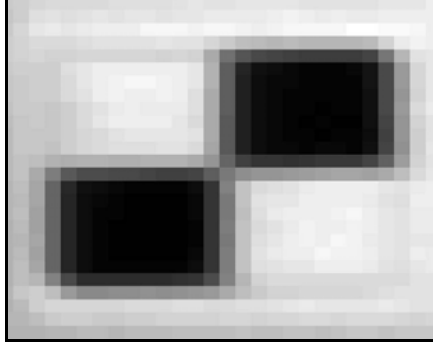


Figure 4a: the target in image patch

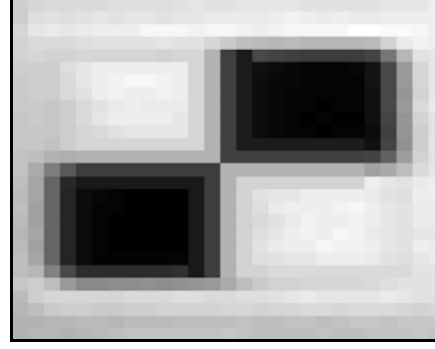


Figure 4b: the ideal or template target

The whole procedure will be repeated as needed but using the new template window with the new intensity values every time and the parameters will be updated. The match will be achieved when the system converges and those 6-parameters do not change any further. The system might diverge if there is no similarity between the two patches or the approximate location of the match far off from the real one by more than several pixels (Mikhail, Moffitt and Francis, 1980; Atkinson, 1996).

4. MATHEMATICAL MODEL FOR SELF-CALIBRATION AND SOLUTION METHOD

4.1 MATHEMATICAL MODEL

The mathematical model was chosen carefully in order to cover all significant sources of geometric errors and estimate all significant correction parameters for those errors. (Samtaney, 1999) explored this model in detail. It was derived from the fundamental collinearity equations. This model relates two coordinate systems to each other. It maps the coordinates from the object space into the image space (Mikhail, bethel and McGlone, 2001). There are two types of parameters. First, the exterior parameters which include the location and orientation parameters. Lens distortion and focal length are examples of the second type, which are called the interior parameters. The model specifically covers and takes into account the lens distortion through some parameters that model radial, decentering, and affinity distortion.

$$\begin{aligned} x - x_o &= x' - x_o + \Delta x \\ &= -f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} \end{aligned} \quad (1a)$$

$$\begin{aligned} y - y_o &= y' - y_o + \Delta y \\ &= -f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} \end{aligned} \quad (1b)$$

Where (x, y) and (x', y') are the ideal and measured target coordinates in image space. x_o, y_o are the principal point coordinates in image space. $\Delta x, \Delta y$ are distortion corrections in x, y directions in image space, see equation (2). f is the camera focal length and $r_{i,j}$ is the i th row and j th column element of the orientation matrix R , see equation (3). (X, Y, Z) are the target coordinates in object space and (X_c, Y_c, Z_c) are exposure station coordinates in object space.

The distortion effects including radial, lens decentering, and affinity were computed through the equations below.

$$\Delta x = \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2 \bar{x} \bar{y} \quad (2a)$$

$$\Delta y = \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 \bar{x} \bar{y} + p_2(r^2 + 2\bar{y}^2) + a_1 \bar{x} + a_2 \bar{y} \quad (2b)$$

In which $\bar{x} = x' - x_o$, $\bar{y} = y' - y_o$ and $r^2 = \bar{x}^2 + \bar{y}^2$. k_i are the radial distortion coefficients, p_i are the decentering distortion coefficients and a_i are the affinity distortion coefficients. The rotation matrix R expresses the orientation of the image coordinate system with respect to the object coordinate system. It is used in the model to relate the target coordinates in both systems. Rotations around the three principle axes x , y , and z by the three angles ω , ϕ , and k generate the matrix R as seen in equation (3).

$$R = R_z(k) R_y(\phi) R_x(\omega) \quad (3)$$

The two condition equations for each target will be:

$$F_x = \bar{x} + \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2 \bar{x} \bar{y} + f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0 \quad (4a)$$

$$F_y = \bar{y} + \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 \bar{x} \bar{y} + p_2(r^2 + 2\bar{y}^2) + a_1 \bar{x} + a_2 \bar{y} + f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0 \quad (4b)$$

From the equations above, each target observation will generate two equations. Consequently, the number of equations will be twice the number of targets in the image for each camera.

4.2 SOLUTION METHOD

The unified least squares approach was used to solve this system since some *a priori* knowledge is available for a number of parameters (Mikhail and Ackerman, 1976). Using the *a priori* knowledge of the parameters is the distinction between ordinary least squares and unified least squares. This *a priori* knowledge is utilized to give those parameters initial values and weights. In this sense, some of the parameters were treated as observations with low precision by assigning large variances to them. Since the system is non-linear, the parameter values will be updated iteratively by adding the correction to them.

The system will converge when the correction vector values are negligible. Then the final correction will be added to the parameters to get the final estimated values. In our case here, the system converges with few iterations since the precision of the observations was very high.

5. DISTORTION ANALYSIS

After determining the camera parameters including distortion parameters, distortion curves were drawn for visual and computational analysis. Radial and decentering (tangential) distortions are discussed in this section.

5.1 RADIAL DISTORTION

The technical term for the displacement of an imaged object radially either towards or away from the principle point is radial distortion (Atkinson, 1996). The magnitude of this displacement (radial distortion) is usually determined to micrometer precision and it varies with the lens focusing. Radial Distortion is included in the math model and its magnitude can be calculated as follows:

$$\Delta r = k_1 r^3 + k_2 r^5 + k_3 r^7 \quad (5)$$

$$\delta_x = \Delta r * \bar{x} / r \quad \delta_y = \Delta r * \bar{y} / r$$

The radial distortion curve was constructed based on the equation above as shown in figure 6. The resulting curves were obtained for all four cameras and the maximum radial distortion was around 30 micrometers.

The following step was done to level or balance the curve based on equalizing the maximum and the minimum distortion values. This procedure is done only to balance the positive and negative excursions of the distortion function about zero. This step has no effect on the final results of the corrected coordinates; it is just cosmetic but accepted professional practice. Mathematically, balancing the curve leads to a change in the radial distortion parameters and consequently the focal length and other related camera parameters. The aim of this balancing procedure is to make $|d_{\max}| = |d_{\min}|$ as shown in figure 6 and the condition equation will be:

$$r_{\max} - CFL \times \tan(\alpha_{\max}) + r_{\min} - CFL \times \tan(\alpha_{\min}) = 0 \quad (6)$$

So the new focal length is

$$CFL = \frac{r_{\max} + r_{\min}}{\tan(\alpha_{\max}) + \tan(\alpha_{\min})} \quad (7)$$

After getting the revised focal length, the calibration adjustment program is run again but with a fixed focal length (CFL). Also, the orientation angles were fixed during this adjustment. This will require adjusting other parameters also to produce the balanced curve in addition to radial distortion parameters k_1 , k_2 and k_3 . The other parameters are principal point shift, decentering and affine distortion parameters. Since these parameters are not independent of the radial distortion parameters, the equalization procedure will be repeated until we get the balanced curve. If all parameters other than the radial distortion parameters are fixed, then the balanced curve will be obtained from one iteration. All of that processing was handled automatically except for the red camera, that curve was balanced manually.

The distortion curves for each iteration of the green band camera are shown in figure 6. Also the radial distortion magnitudes and their orientations throughout the image plane are shown along with the principal point, PPS, and the fiducial center, FC, in figure 7.

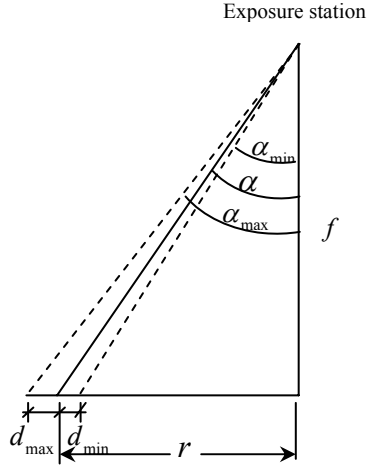


Figure 5: image plane cross section.

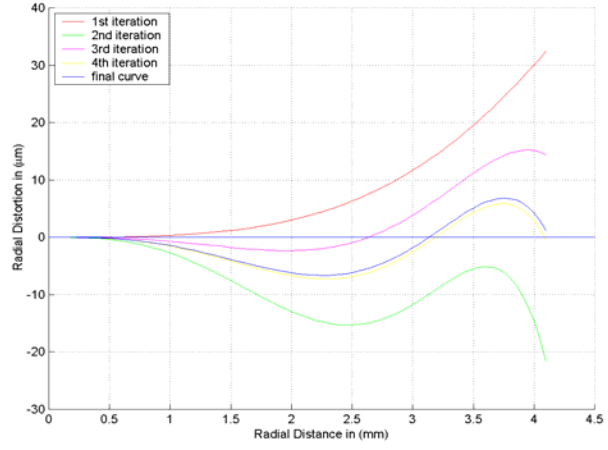


Figure 6: radial distortion for the green camera in CAMIS sensor

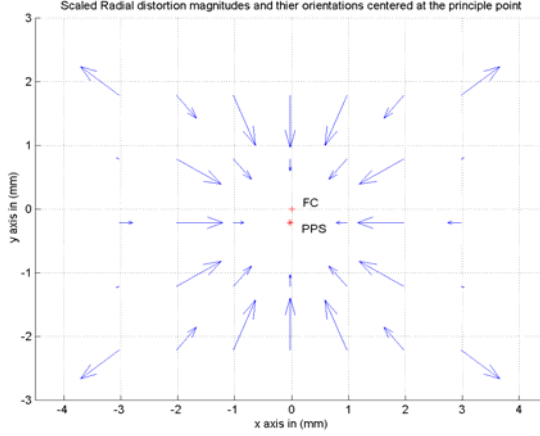


Figure 7a: Scaled radial distortion on image plane centered at PPS for the green camera

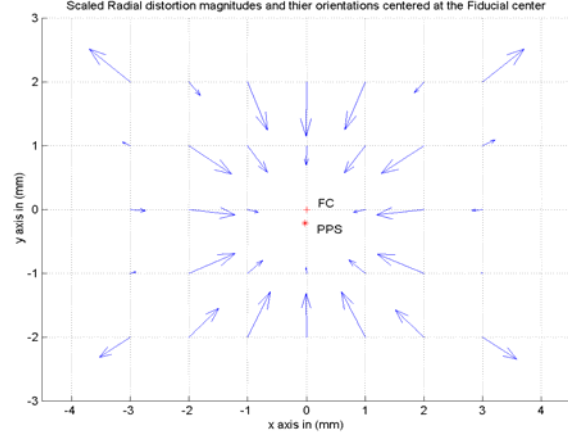


Figure 7b: Scaled radial distortion on image plane centered at FC for the green camera

5.2 DECENTERING DISTORTION

When a lens is manufactured, all its components should be aligned perfectly. But such perfection is not possible. The misalignment will lead to systematic image displacement errors. This undesired geometric displacement in the image is called decentering distortion. In this calibration procedure the following mathematical model is used (Samtaney, 1999; Atkinson, 1996),

$$\begin{aligned}\delta_x &= p_1[r^2 + 2(x - x_o)^2] + 2p_2(x - x_o)(y - y_o) \\ \delta_y &= p_2[r^2 + 2(y - y_o)^2] + 2p_1(x - x_o)(y - y_o)\end{aligned}\quad (8)$$

As mentioned earlier, the equalization procedure for the radial distortion has an affect on the other parameters. So, in each equalization iteration, the decentering parameters will have new values since their behavior will be adjusted according to the modification of the focal length. Nevertheless, the equalization technique does not change the final corrected coordinate values and the main purpose for it is to make the distortion correction balanced in magnitude. Figure 8 shows the scaled magnitudes of the decentering distortions and their orientations throughout the image plane with respect to the principal point (PPS) and the fiducial center (FC) of the image.

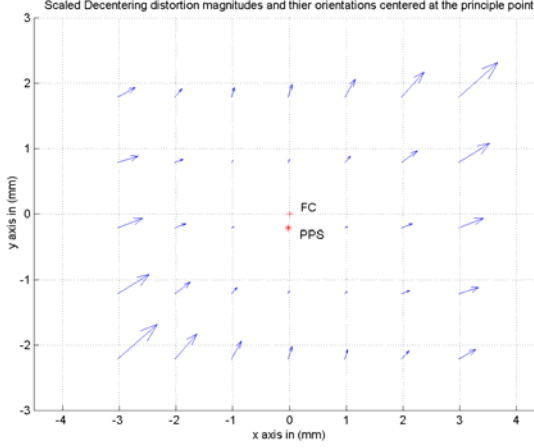


Figure 8a: Scaled Decentering Distortion on image plane centered at PPS for the Green camera

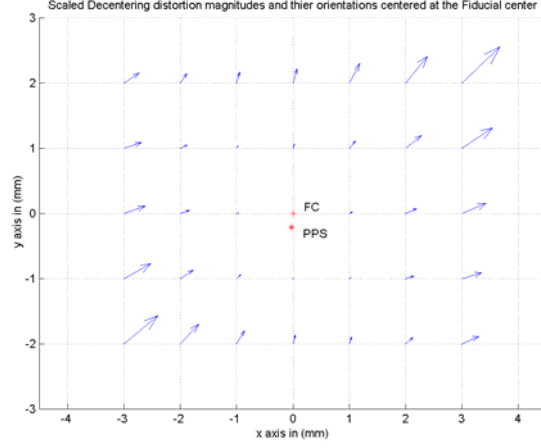


Figure 8b: Scaled Decentering Distortion on image plane centered at FC for the Green camera

6. Results and Discussion

The resulting calibration parameters for the four cameras are summarized in the table below. Those parameters can be used to refine the coordinate observations in image space for each camera, respectively. We tried during this work to automate the calibration process as much as possible. We anticipate that this careful calibration will improve the results from bundle block adjustment using the CAMIS sensor. Verification of this will have to await more testing. The interesting contributions of this research have been, the setup and measurements of the targets and the cameras, the automation of the target locations in the images, and their subsequent refinement, and the automatic process for balancing the radial lens distortion in the presence of other correlated parameters.

Parameter	Blue Camera Working band 450nm	Green Camera Working band 550nm	Red Camera Working band 650nm	Alpha Camera Working band 800nm
f	16.168 mm	16.154 mm	16.177 mm	16.174 mm
x_o	0.084357 mm	-0.024693 mm	-0.253464 mm	-0.014537 mm
y_o	-0.239466 mm	- 0.213401 mm	- 0.109878 mm	-0.158923 mm
k_1	-1.696019×10^{-3}	-1.709735×10^{-3}	-2.011666×10^{-3}	-1.712994×10^{-3}
k_2	0.257710×10^{-3}	0.274246×10^{-3}	0.337008×10^{-3}	0.260639×10^{-3}
k_3	-0.009089×10^{-3}	-0.010202×10^{-3}	-0.013096×10^{-3}	-0.009190×10^{-3}
p_1	0.053187×10^{-3}	0.036416×10^{-3}	-0.134155×10^{-3}	-0.009700×10^{-3}
p_2	0.113262×10^{-3}	0.036419×10^{-3}	0.066029×10^{-3}	0.014206×10^{-3}
a_1	1.162203×10^{-3}	0.279185×10^{-3}	-0.059959×10^{-3}	0.130802×10^{-3}
a_2	0.059299×10^{-3}	-0.663338×10^{-3}	-0.637141×10^{-3}	0.278204×10^{-3}

Table 1: Estimated parameters of the four sensors.

ACKNOWLEDGEMENT

The authors would like to acknowledge the support of the Army Research Office and the Topographic Engineering Center.

REFERENCES

- Atkinson, K. B., (1996). *Close Range Photogrammetry and Machine Vision*.
- Batista, Jorge., Araujo, Helder., Almeida, A. T., (1998). Iterative Multi-Step Explicit Camera Calibration, 1998 *IEEE Int. Conf. on Computer Vision (ICCV98)*, Bombay, India, January 4-7.
- Bethel, James S., Mikhail, Edward M., (2001). *Introduction to Modern Photogrammetry*.
- Mikhail, Edward M., Ackermann, F., (1976). *Observation and Least Square*.
- Moffit, Francis H., Mikhail, Edward M., (1980). *Photogrammetry*, third edition.
- Pollefeys, Mark., Koch, Reinhard., Gool, Lue Van. (1999). Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters, *International Journal of Computer Vision*, 32(1), 7-25.
- Ravi Samtaney. (1999). A Method to Solve Interior and Exterior Camera Calibration Parameters for Image Resection, *NAS-99-03*.

Geometric Calibration Report of CAMIS Sensor

For

Topographic Engineering Center

By

A. Alharthy & J. Bethel

School of Civil Engineering

Purdue University

25 MAY, 2001

Table of Content

1. Abstract:	3
2. Calibration:	4
2.1 Site preparation:	4
2.2 Measurements and adjustment in object space:	5
2.3 Capturing Images and obtaining image space coordinates:	7
2.4 Image space calculations:	7
2.5 Camera parameters estimation:	8
3 Results:	9
3.1 Blue Camera:	9
3.2 Green Camera:	10
3.3 Red Camera:	11
3.4 Near Infra Red:	12
Appendix A: Target Locations in image space (Matching):	13
A.1 Introduction:	13
A.2 Cross-correlation matching:	13
A.3 Least squares matching (LSQM):	14
Appendix B: Math Model	18
B.1 Math model	18
B.2 Solution method	20
B.3 Numerical example:	21
Appendix C: Distortion Analysis:	24
C.1 Radial Distortion:	24
C.2 Decentering Distortion:	26
REFERENCES	40

1. Abstract:

The CAMIS sensor consists of four co-baresighted area-CCD cameras with filters to scan. The scan radiance in the blue, green, red, and near infrared spectral bands. In this report, we summarize the work that has been done during the geometric calibration of the CAMIS sensor. The procedure required many preliminary steps such as preparing the calibration site which involved target layout, setting up the coordinate system and locating objects to that system. Theodolites and a steel tape were used to measure the angles and distances in the network of calibration targets. In order to adjust those measurements and to get the target coordinates in the reference coordinate system, we developed network bundle adjustment program. Images of the target field were then taken by each of the four cameras of the CAMIS sensor. The coordinates of the targets in both the object and the image system were used as observations for estimating the sensor parameters. Two matching techniques were used to determine and refine the target locations in the image space. We modified the matching algorithm to overcome certain radiometric effects and thereby found the location of the target centers in image space.

After planning the data flow, the images were taken and the calibration procedure was started. To cover the most significant parameters, a full math model was used. A description of this model and its use are attached to this report. The Unified least square approach was used iteratively to solve this nonlinear over determined system since we have some prior knowledge about a number of the sensor parameters. Finally, in order to see the distortion behavior, the radial and decentering distortions were calculated along the radial distance and plotted separately. Then the radial distortion curve was equalized and the corresponding changes to the sensor parameters were recorded. We repeated the procedure for each camera individually and consequently our results, which are presented in this report, have four sets of adjusted parameters, one per camera. In the appendices, the report also outlines the basic steps and algorithms that were used during the calibration process.

2. Calibration:

The aim of this work was to make a laboratory calibration for the geometric parameters of the CAMIS sensor. CAMIS stands for Computerized Airborne Multicamera Imaging System. It is very advanced system and is designed to utilize modern solid-state imaging and data acquisition technology. It is composed of four CCD cameras with band pass optical filters to obtain four band images. The center wavelength of those bands is as follows: 450, 550, 650



Figure 1a: the four cameras

and 800 nm. However, each sensor has its own optics and obtains its own image based on its working band independently from the others at the same time. Those four images can be integrated into a composite image or viewed individually. The CAMIS sensor has been used in multispectral imaging and mapping purposes by mounting it on an airplane with GPS and INS systems. Moreover, the PC that is used with this sensor is capable of recording and managing simultaneous image capture.



Figure 1b: CAMIS sensor

The procedure required a number of steps and they are summarized below:

2.1 Site preparation:

The idea of the calibration was to layout some targets in the object space, locate them accurately, and acquire an image of those targets by the sensor. Then, we relate the coordinates of the targets in both systems, image and object space, in order to obtain the camera parameters. So, the procedure

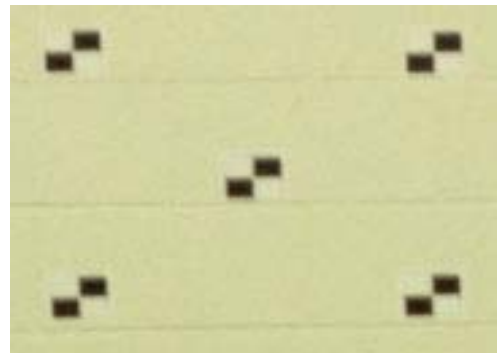


Figure 2: targets shape

starts by setting up the calibration site. First, we designed the targets to be in cross

shapes so their center positions will be read very easily. Then they have been laid out to be distributed in the image scene in a specific way that enables us to cover most geometric and systematic errors as shown in [figure 3](#). Those targets were placed on an almost flat service wall and the sensor position was located around 8 meters away from that wall.

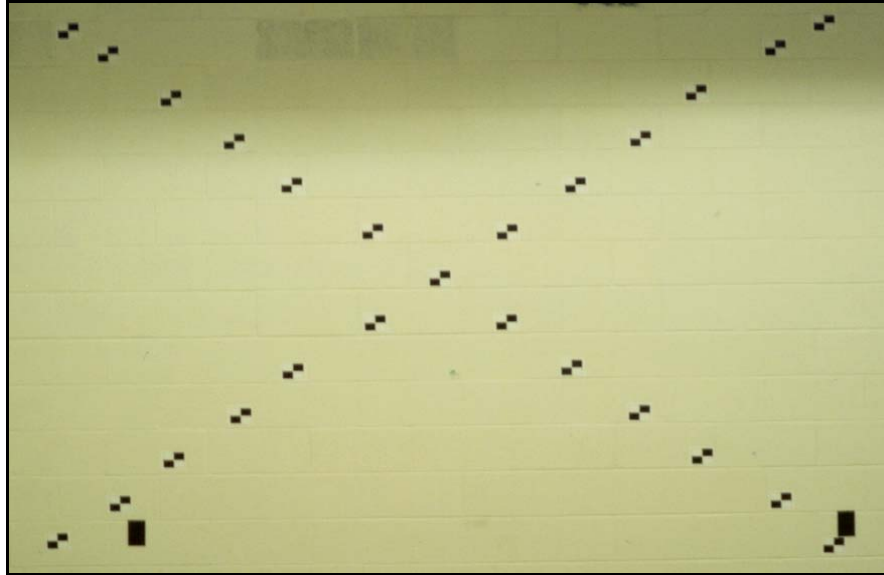


Figure 3: targets layout

In order to register the objects in the scene, an object space coordinate system was established at the site and two other instrument stations were marked to use in the measurements.

2.2 Measurements and adjustment in object space:

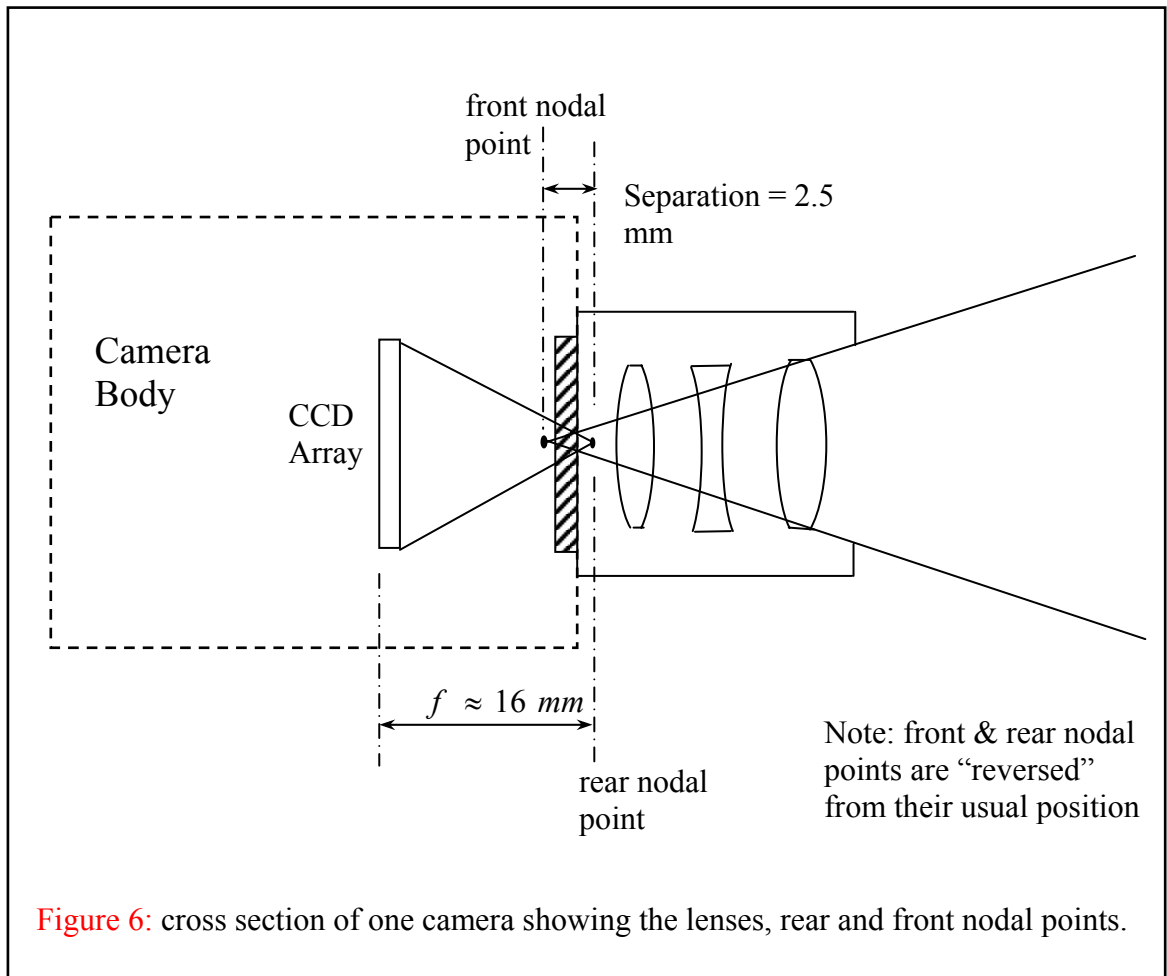
Two 3 arc-seconds accuracy theodolites were mounted at these two stations and were used to measure the angles between the targets themselves on one hand and between the targets and the four exposure stations of the CAMIS cameras on the



Figure 4: measuring angles

other hand. The origin of the object space was chosen to be theodolite 1 at station one in the right of the sensor position. Many manual measurements were made of the

camera physical layout, using machinist calipers. The lenses were also placed on an optical bench for determine the locations of the nodal points. A cross section of one camera is shown in [figure 6](#). This was needed to locate the camera front nodal point with respect to the camera body which would be located in the network by theodolite observations. The spacings between the targets were measured with a steel tape. Having all these observations, we end up with an over determined system of equations. So, we developed a bundle adjustment program to adjust those coordinates of the targets and the instrument stations. As a result of that, we determined precisely all our targets and stations in the object space and the next step was capturing images by the sensor.



2.3 Capturing Images and obtaining image space coordinates:

Before the measurements, the sensor was mounted on leveled plate over a tripod. In this sense, the exposure stations were fixed and accurately determined. The image-capturing step was operated by a team from TEC and observed by the Purdue team. In this step we tried to simulate the real working conditions by setting the lenses to the infinity focus position. Images were viewed at the site to make sure that as many as possible of the targets were exposed. After this step, the site work ended and the processing procedure started.

2.4 Image space calculations:

Once the images were captured, we ran a cross correlation matching code to get rough approximation of target positions in the image space to within a few pixels. The cross correlation matching function works by computing the similarity between two same sized widows. One window patch contains the ideal target shape and the other contains a window from the image. In general, a matching problem is a key algorithm for other applications and image analysis. This function is explored more in [appendix A2](#). Despite the fact that, the cross correlation matching results showed that we are only away from the exact position by a pixel or less, we needed more accurate and precise methods to define the exact location with a hundredth of a pixel or so. This step is very essential for the coming refinement step. This level of accuracy is mandatory for a camera calibration problem. Least Square matching is very adequate technique for this purpose and very powerful. LSQ matching utilizes the first derivative (gradient) of the intensity in both directions x,y to obtain the best match and the exact matching can be reached by moving one window towards the other one. Some obstacles were faced and solved by modifying the algorithm, see



Figure 5: Target's center using positioning using two matching algorithms.

[appendix A3](#) for more details. As a result of this step, the image space coordinates for the targets have been obtained with a fine accuracy. Results using both techniques are shown in [figure 5](#).

2.5 Camera parameters estimation:

First, a full math model was chosen to perform the geometric calibration and estimate the sensor parameters. In this math model, Ravi Samtaney(1999) tried to cover most factors that might occur in geometric calibration. This model is explored in more details in [appendix B](#). The model relates the target coordinates in object space and image space through the camera parameters. This overdetermined and non-linear system needs an optimization criterion to be solved. Since we have some initial values for a number of the camera parameters and their ranges, we decided to use the Unified Least Square algorithm iteratively to solve the system. Using the resulting distortion parameters, plots were drawn to describe the radial and decentering distortion behavior. Finally, the radial distortion curve **was equalized** by small changes in corresponding parameters.

The results of the process are tabulated for each sensor individually. Also, a numerical example is illustrated in [appendix C](#) to demonstrate the use of those parameters.

3 Results:

3.1 Blue Camera:

- Sensor type: working band (450 nm) ‘Blue band’.

Camera Model: CAMIS Model 4768P

Sensor type: Sony XC-8500 CE 1/2” CCD Progressive Scan Camera.

Lens type: TV Lens made by Cosmocar Pentax, Japan.

Shutter speed: 1/50 up to 1/10,000 sec.

Spec. CCD array size: 782 x 582 pixel.

Actual CCD array size: 768 x 576 pixel.

Nominal Focal Length: 16.000 mm.

- Estimated Parameters for the sensor:

Parameter symbol	Calibrated value	Short Description
f	16.168 mm	Calibrated Focal Length “EFL”
x_o	0.084357 mm	Principal Point x coordinate
y_o	- 0.239466 mm	Principal Point y coordinate
k_1	$-1.696019 * 10^{-3}$	Radial Distortion parameters
k_2	$0.257710 * 10^{-3}$	
k_3	$-0.009090 * 10^{-3}$	
p_1	$0.053188 * 10^{-3}$	Lens decentering parameters
p_2	$0.113262 * 10^{-3}$	
a_1	$1.162204 * 10^{-3}$	Affinity Distortion parameters
a_2	$0.059300 * 10^{-3}$	

- See math model for more details.
- Conversion factor: 1 pixel = 0.0083 mm.

3.2 Green Camera:

- Sensor type: working band (550 nm) ‘Green band’.

Camera Model: CAMIS Model 4768P

Sensor type: Sony XC-8500 CE 1/2” CCD Progressive Scan Camera.

Lens type: TV Lens made by Cosmocar Pentax, Japan.

Shutter speed: 1/50 up to 1/10,000 sec.

Spec. CCD array size: 782 x 582 pixel.

Actual CCD array size: 768 x 576 pixel.

Nominal Focal Length: 16.000 mm.

- Estimated Parameters for the sensor:

Parameter symbol	Calibrated value	Short Description
f	16.154 mm	Calibrated Focal Length “EFL”
x_o	-0.024693 mm	Principal Point x coordinate
y_o	- 0.213401 mm	Principal Point y coordinate
k_1	$-1.709735 * 10^{-3}$	Radial Distortion parameters
k_2	$0.274246 * 10^{-3}$	
k_3	$-0.010202 * 10^{-3}$	
p_1	$0.036417 * 10^{-3}$	Lens decentering parameters
p_2	$0.036419 * 10^{-3}$	
a_1	$0.279185 * 10^{-3}$	Affinity Distortion parameters
a_2	$-0.663339 * 10^{-3}$	

- See math model for more details.
- Conversion factor: 1 pixel = 0.0083 mm.

3.3 Red Camera:

- Sensor type: working band (650 nm) ‘Red band’.

Camera Model: CAMIS Model 4768P

Sensor type: Sony XC-8500 CE 1/2” CCD Progressive Scan Camera.

Lens type: TV Lens made by Cosmocar Pentax, Japan.

Shutter speed: 1/50 up to 1/10,000 sec.

Spec. CCD array size: 782 x 582 pixel.

Actual CCD array size: 768 x 576 pixel.

Nominal Focal Length: 16.000 mm.

- Estimated Parameters for the sensor:

Parameter symbol	Calibrated value	Short Description
f	16.195 mm	Calibrated Focal Length “EFL”
x_o	-0.252689 mm	Principal Point x coordinate
y_o	- 0.108910 mm	Principal Point y coordinate
k_1	$-1.490695 * 10^{-3}$	Radial Distortion parameters
k_2	$0.267886 * 10^{-3}$	
k_3	$-0.010391 * 10^{-3}$	
p_1	$-0.140914 * 10^{-3}$	Lens decentering parameters
p_2	$0.057370 * 10^{-3}$	
a_1	$-0.062786 * 10^{-3}$	Affinity Distortion parameters
a_2	$-0.608008 * 10^{-3}$	

- See math model for more details.
- Conversion factor: 1 pixel = 0.0083 mm.

3.4 Near Infra Red:

- Sensor type: working band (800 nm) ‘Near Infra Red band’.

Camera Model: CAMIS Model 4768P

Sensor type: Sony XC-8500 CE 1/2” CCD Progressive Scan Camera.

Lens type: TV Lens made by Cosmocar Pentax, Japan.

Shutter speed: 1/50 up to 1/10,000 sec.

Spec. CCD array size: 782 x 582 pixel.

Actual CCD array size: 768 x 576 pixel.

Nominal Focal Length: 16.000 mm.

- Estimated Parameters for the sensor:

Parameter symbol	Calibrated value	Short Description
f	16.173 mm	Calibrated Focal Length “CFL”
x_o	-0.014533 mm	Principal Point x coordinate
y_o	-0.158918 mm	Principal Point y coordinate
k_1	$-1.739689 * 10^{-3}$	Radial Distortion parameters
k_2	$0.263932 * 10^{-3}$	
k_3	$-0.009311 * 10^{-3}$	
p_1	$-0.009317 * 10^{-3}$	Lens decentering parameters
p_2	$0.014705 * 10^{-3}$	
a_1	$0.131437 * 10^{-3}$	Affinity Distortion parameters
a_2	$0.276288 * 10^{-3}$	

- See math model for more details.
- Conversion factor: 1 pixel = 0.0083 mm.

Appendix A: Target Locations in image space (Matching)

A.1 Introduction

In order to find the exact target locations in the image, two matching approaches were used. First, the approximate locations are obtained using the cross correlation matching. Second, we refine the results of the first algorithm using least squares matching.

A.2 Cross-correlation matching

Cross-correlation determines the similarity between two corresponding patches. The cross-correlation approach cannot give the precise location of an object due to many factors. Differences in attitude, distortion, and signal noise are some examples that affect the correspondence in geometry and radiometry. However, this algorithm usually gives an approximate location of the correspondence within a few pixels.

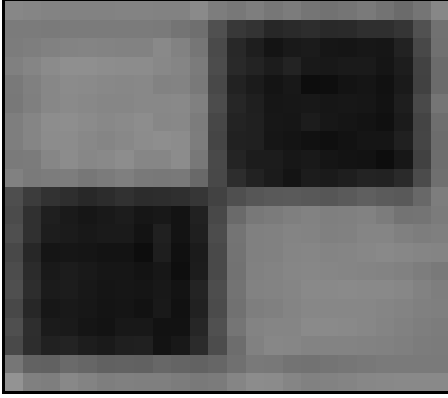


Figure A1: image patch (v)



Figure A2: ideal template (u)

The cross-correlation matching function over two windows of N pixels each can be written as:

$$C_{uv} = \frac{\sum_{i=1}^N (u_i - \bar{u}) (v_i - \bar{v})}{\left[\sum_{i=1}^N (u_i - \bar{u}) \sum_{i=1}^N (v_i - \bar{v}) \right]^{1/2}} \quad \text{eq(A1).}$$

Where: C_{uv} : match function between the template window (u) and image window (v).

u_i, v_i : intensity value at position i .

\bar{u}, \bar{v} : mean intensity value of the window $\frac{\sum_{i=1}^N u_i}{N}$, $\frac{\sum_{i=1}^N v_i}{N}$ respectively.

The ideal template will be passed through the image and the matching function will be computed and recorded at the center pixel of the patch. The match function ranges between +1 and -1. The max value equals +1 which means there is a full match between the two windows or, in other words, they are identical. Usually a threshold will be used to distinguish between matches and non-matches. Then the pixels with a cross correlation above the threshold will be considered candidates for the match. A copy of the developed program for this procedure is in the accompanying CD and some patches of the results are shown below. Our results, as shown in figure A3, are off from the center of the target only by a pixel or less. These results enable us to use the least squares matching/refining technique directly.

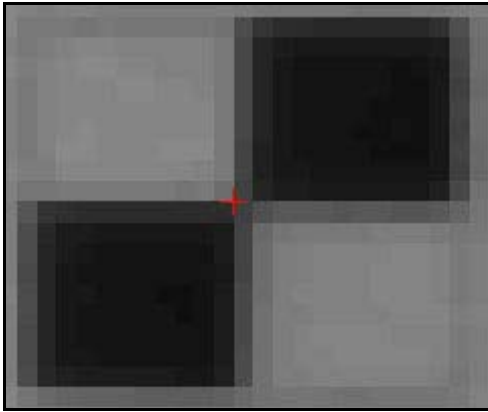


Figure A3a: a cross_corr. Match

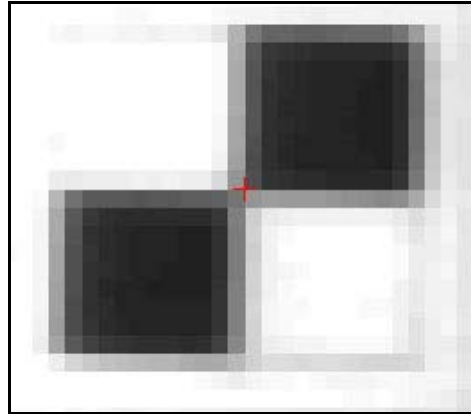


Figure A3b: another cross_corr. match

A.3 Least squares matching (LSQM):

Least squares matching works as a powerful and an accurate technique to refine an object's coordinates in the image space based on the correspondence between two relatively small windows of two images. This technique uses the gradient in the x,y directions in order to move the two patches towards each others to get the perfect match. The match precision that we are looking for with this technique is within a hundredth of a pixel or so. The similarity between the two spots was only geometrically modeled for this specific problem since the radiometric effect was eliminated through some preprocessing steps, as we will see below. The problem is to match an ideal shape of the target with a small window from the image containing the imaged target. The following steps describe the procedure for setting up the two windows for matching:

- i) Get the approximate location of the imaged target using the first matching approach (cross correlation) as described above in appendix A2. Those locations should be within a few pixels of the exact location in order to make the geometric model in LSQM converge and to produce accurate results.
- ii) Having the rough estimated location, a window around that location from the image with adequate size will be extracted for matching purposes. This is all done systematically inside the code.
- iii) Then, the pixel gray level values in this window will be used to assemble the ideal target for matching as shown in figure A4. In the extracted patch, the center of the target can be seen very clearly while in the image patch it not clear. Moreover, the similarity in the intensity is preserved between the two windows. Also, we tried to preserve the gradient by gradually distributing the change in the intensity between the black and gray regions. In this sense the radiometric effect on the correspondence will be diminished.

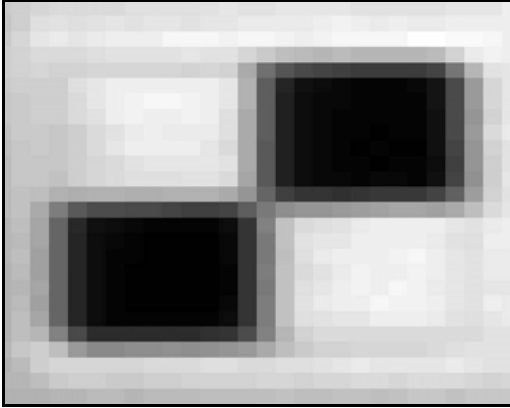


Figure A4a: the target in Image patch

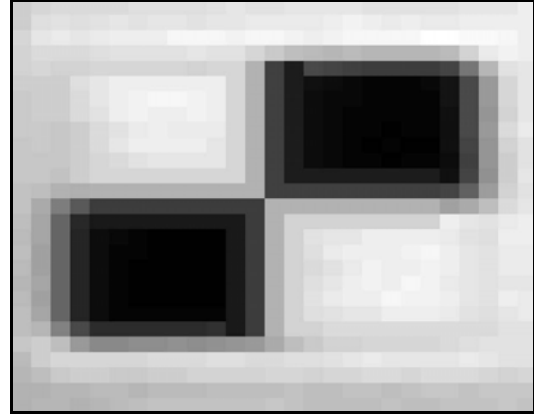


Figure A4b: the extracted target from Image Patch to use it as the ideal target in matching

After specifying the two windows with the same size for matching the LSQM procedure will take a place. Requiring equality in intensity between each of the two corresponding pixels from the two windows is the basic condition for this procedure as shown below.

$$F = img(i, j) - temp(i, j) = 0 \quad \text{eq(A2).}$$

Where ‘img’ represent the patch that was taken from image in order to match it with ‘temp’, which represents the ideal template.

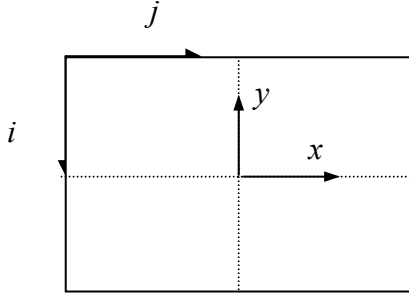


Figure A5a: Image patch

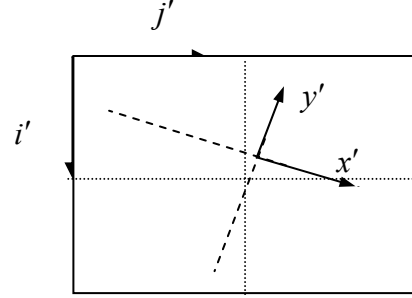


Figure A5a: ideal template

x, y and x', y' are not in the same coordinate system. A transformation, which consists of 6-parameters, is used to link those two systems.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_o \\ b_o \end{bmatrix}$$

in which a_1, a_2, b_1, b_2 are the elements of the transformation matrix R and a_o, b_o are the elements of the shift vector t . Then the linearized form of the least square is used to estimate those parameters.

$$v + B \Delta = f \quad \text{eq(A3),}$$

Where v is a vector of the residuals to the parameters and B is a matrix consists of the partial derivatives of the condition equation with respect to the parameters as shown below. Δ is the correction vector to those parameters and $f = -F$.

$$B = \begin{bmatrix} \frac{\partial F}{\partial a_1} & \frac{\partial F}{\partial a_2} & \frac{\partial F}{\partial a_o} & \frac{\partial F}{\partial b_1} & \frac{\partial F}{\partial b_2} & \frac{\partial F}{\partial b_o} \end{bmatrix}$$

taking the derivatives via the chain rule, the above expression will be:

$$\frac{\partial F}{\partial a_1} = - \frac{\partial temp}{\partial a_1} = - \frac{\partial temp}{\partial x'} \frac{\partial x'}{\partial a_1}$$

in which $\frac{\partial temp}{\partial x'} = g_x$ and $\frac{\partial x'}{\partial a_1} = x$

The first part of the each partial derivative in B (g_x and g_y) was estimated by the gradients in the x and y directions respectively as follows:

$$g_y = - \frac{temp(i+1, j) - temp(i-1, j)}{2}, \quad g_x = \frac{temp(i, j+1) - temp(i, j-1)}{2}$$

Consequently the matrix B will be:

$$B_{(k,6)} = \begin{bmatrix} -g_x x & -g_x y & -g_x & -g_y x & -g_y y & -g_y \end{bmatrix}$$

From the above drawing in figure 5,

$$x = j - (size_{patch} / 2) ; \quad y = -(i - (size_{patch} / 2))$$

and

$$f_{(k,1)} = -(img(i, j) - temp(i, j)) ,$$

B and f will be calculated for each pixel and its corresponding pixel in the other patch.

The two patches that we should work with should be the same size (m,n). So, $k = m * n$.

Using all pixels (k), the least squares technique will estimate the 6-parameters for the first iteration, assuming unit weights,

$$\Delta = inv(B' * B)(B' * f)$$

This correction vector will be added to the 6-parameter initial values. Those corrected parameters will be used to calculate the new coordinates x', y' in order to use them in resampling the grid for the template window. We used bilinear interpolation to estimate the intensity values during this resampling procedure.

The whole procedure will be repeated as needed but using the new template window with the new intensity values every time and the parameters will be accumulatively updated. The match will be achieved when the system converges and those 6-parameters do not change any further. The system might diverge if there is no similarity between the two patches or the approximate location of the match far off from the real one by more than several pixels. For more details, see the matlab LSQM matching program that is in the accompanying CD.

For the purpose of comparison, the match from cross-correlation and LSQM are viewed in figure A6. We can see clearly how the match moved towards the right center of the targets using LSQM.

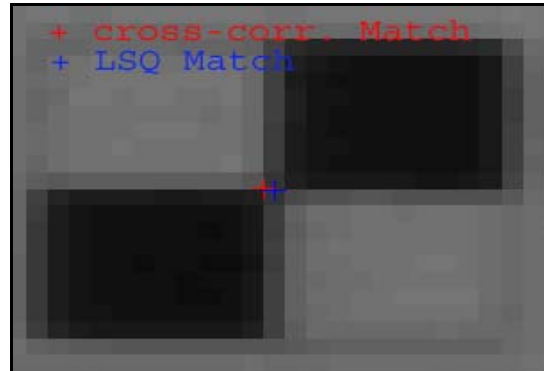


Figure A6: matching results using cross-Correlation and LSQ matching techniques.

Appendix B: Math Model

B.1 Math model

The math model was chosen carefully in order to cover all possible sources of geometric errors and estimate all significant correction parameters for those errors as in [eq B1](#). Samtaney (1999) explored this model in detail. It was derived from the fundamental collinearity equations. This model relates two coordinate systems to each other. It maps the coordinates from the object space into the image space and vice versa taking in account that it is unique only in one direction. There are two types of parameters. First, the *extrinsic* parameters which include the location and orientation parameters. Lens distortion and focal length are examples of the second type, which are called the *intrinsic* parameters. The model specifically covers and takes into account the lens distortion influence through some factors such as radial, decentering, and affinity distortion see [eq B2](#).

$$\begin{aligned} x - x_o &= x' - x_o + \Delta x \\ &= -f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} \end{aligned} \quad \text{eq B1a}$$

$$\begin{aligned} y - y_o &= y' - y_o + \Delta y \\ &= -f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} \end{aligned} \quad \text{eq B1b}$$

$(x, y); (x', y')$: ideal and measured targets projected coordinates in image space.

x_o, y_o : principal point coordinates in image space.

$\Delta x, \Delta y$: distortion correction in x, y directions in image space, see [eq B2](#).

f : camera focal length.

$r_{i,j}$: the i th row and j th column element of the orientation matrix R , see [eq B3](#).

(X, Y, Z) : target coordinates in object space.

(X_c, Y_c, Z_c) : exposure station coordinates in object space.

The distortion effects including radial, lens decentering, and affinity were computed through the equations below.

$$\begin{aligned}\Delta x &= \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2\bar{x}\bar{y} \\ \Delta y &= \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1\bar{x}\bar{y} + p_2(r^2 + 2\bar{y}^2) + a_1\bar{x} + a_2\bar{y} \quad \text{eq B2}\end{aligned}$$

in which: $\bar{x} = x' - x_o$, $\bar{y} = y' - y_o$, $r^2 = \bar{x}^2 + \bar{y}^2$.

k_i : radial distortion coefficients.

p_i : decentering distortion coefficients.

a_i : affinity distortion coefficients.

The rotation matrix R expresses the orientation of the image coordinate system with respect to the object coordinate system. It is used in the model to relate the target coordinates in both systems. Rotations around the three principle axes x , y , and z by the three angles ω , ϕ , and k generate the matrix R as seen in eq B3.

$$R = R_z(k) R_y(\phi) R_x(\omega) \quad \text{eq B3}$$

and those rotations are given by:

$$R_x(\omega) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix}, R_y(\phi) = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}, R_z(k) = \begin{bmatrix} \cos k & \sin k & 0 \\ -\sin k & \cos k & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The two condition equations for each target will be:

$$\begin{aligned}Fx &= \bar{x} + \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2\bar{x}\bar{y} \\ &+ f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0 \quad \text{eq B4a}\end{aligned}$$

$$\begin{aligned}Fy &= \bar{y} + \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1\bar{x}\bar{y} + p_2(r^2 + 2\bar{y}^2) + a_1\bar{x} + a_2\bar{y} \\ &+ f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0 \quad \text{eq B4b}\end{aligned}$$

From the equations above, each target will form two equations. Consequently, the number of equations will be twice the number of targets in the image. So, we can conclude that the number of calibration targets in each image should be at least half of the number of camera parameters that we want to estimate in order to solve this system. However, in a calibration process like this we want the number of equations to exceed the minimum requirement in order to have an *overdetermined* system. Such a system enhances the reliability of the result. In this problem, 13 parameters were estimated and the number of targets that were used was 20 and 21 in some cases. So, the redundancy we had during the procedure was 27.

B.2 Solution method

The Unified least squares approach was used to solve this system since some *a priori* knowledge is available for a number of parameters. Using the a priori knowledge of the parameters is the distinction between the ordinary least squares and the unified least squares. This a priori knowledge is utilized to give those parameters initial values and weights. In this sense, some of the parameters were treated as observations with low precision by assigned large variances to them. Since the system is non-linear, the parameter values will be updated iteratively by adding the correction vector Δ , to them.

$$\Delta = (N + W_{xx})^{-1} * (t - W_{xx} f_x) \quad \text{eq B5}$$

in which,

$$\begin{aligned} N &= B' * (A * Q * A')^{-1} * B \\ t &= B' * (A * Q * A')^{-1} * f_o \\ f_o &= -(F + A * (L - L_o)) \end{aligned} \quad \text{eq B6}$$



Geometric Calibration of the CAMIS Sensor

Abdullatif Alharthy & James Bethel
Geomatics Area, School of Civil Engineering
Purdue University
May 6, 2002



Presentation Outline

- 1) Introduction:
 - Research Objectives
 - Sensor Description
- 2) Laboratory work:
 - Site preparation
 - Measurements and adjustment in object space
 - Capturing Images
- 3) Target Locations in image space (Matching):
 - Cross-correlation matching (CCM)
 - Least squares matching (LSQM)



- 4) Mathematical Model:
 - Mathematical model
 - Solution method
- 5) Distortion Analysis:
 - Radial Distortion
 - Decentering Distortion
- 6) Results and Discussion



1.1 Research Objectives

- CAMIS sensor has been used in multispectral imaging and mapping purposes by mounting it in an airplane with GPS and INS systems.
- These auxiliary sensors provide very good position and attitude data for stabilizing the subsequent bundle block adjustment.
- The aim of this work was to make a laboratory calibration for the geometric parameters of the CAMIS sensor.



1.2 Sensor Description

- CAMIS stands for Computerized Airborne Multicamera Imaging System.
- CAMIS is a system for airborne remote sensing and is designed to utilize modern solid-state imaging and data acquisition technology.
- It consists of four co-boresighted area-CCD cameras with band pass filters: blue, green, red, and near infrared.
- Each sensor has its own optics and obtains its own image independently from the others at the same time.
- The sensor is operated by Topographic Engineering Center and we have worked with Mr. Mitch Pierson to obtain the required data.

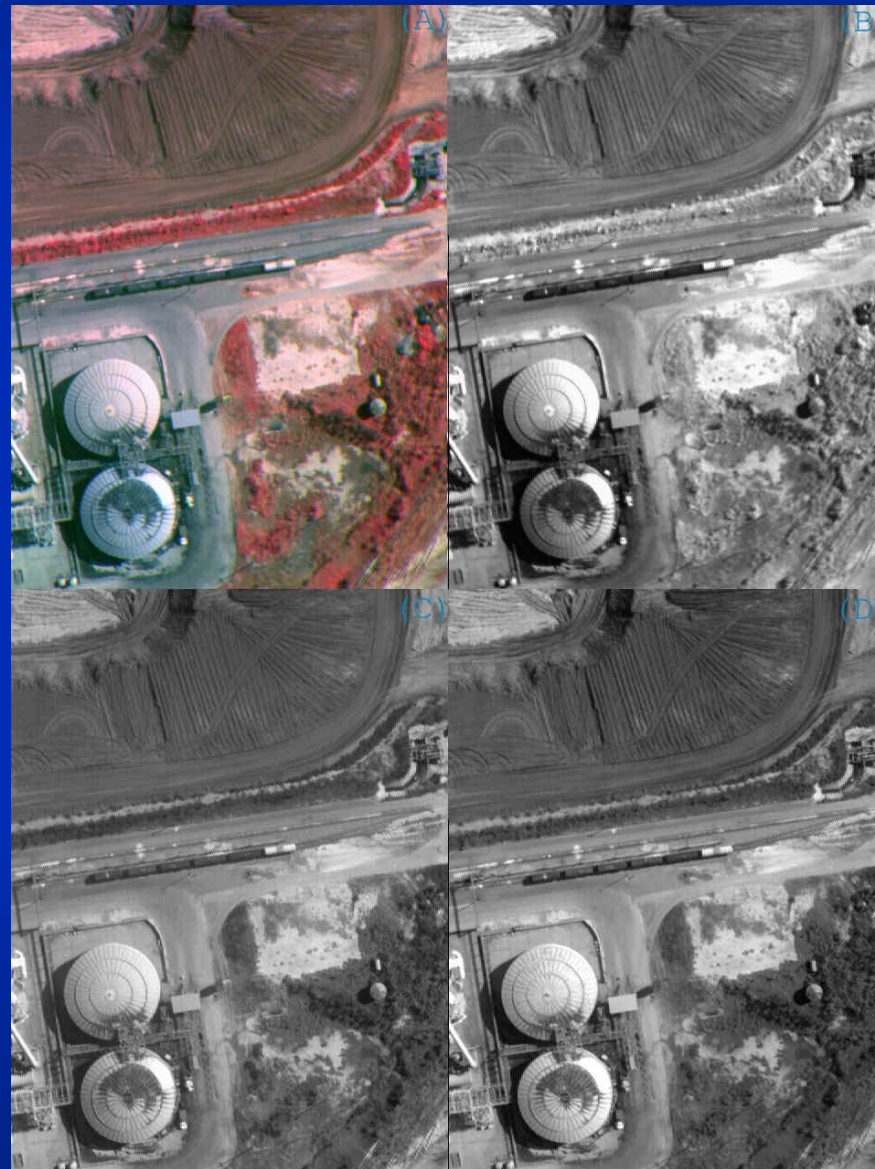


CAMIS, the four cameras

CAMIS in Aircraft with Image Display

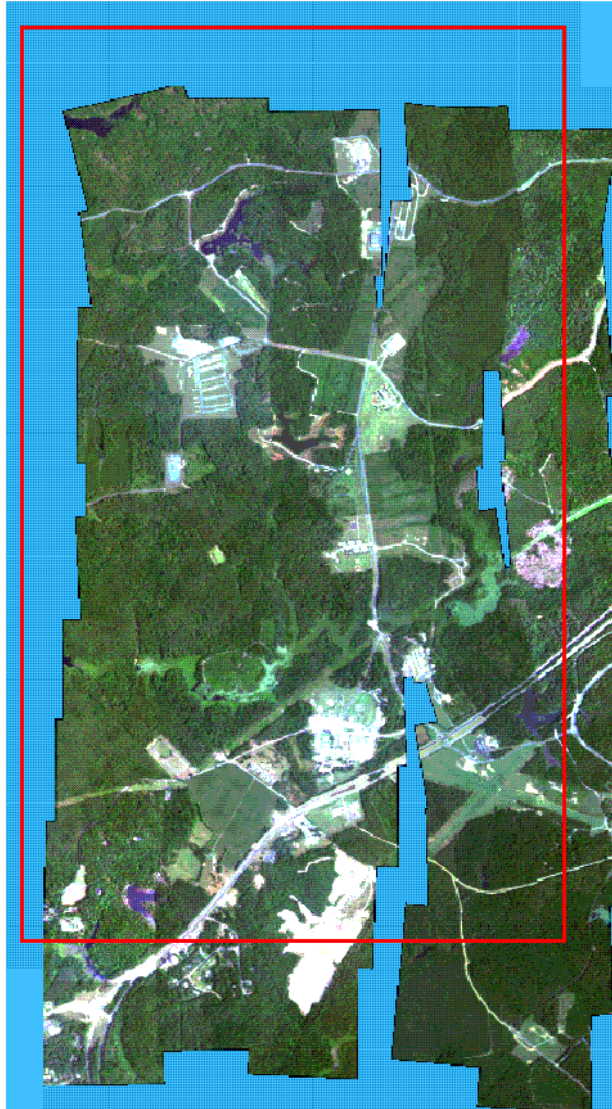


CAMIS 3 Bands and Color Composite



Mosaic from CAMIS Strips

Ft. A.P. Hill Mosaic





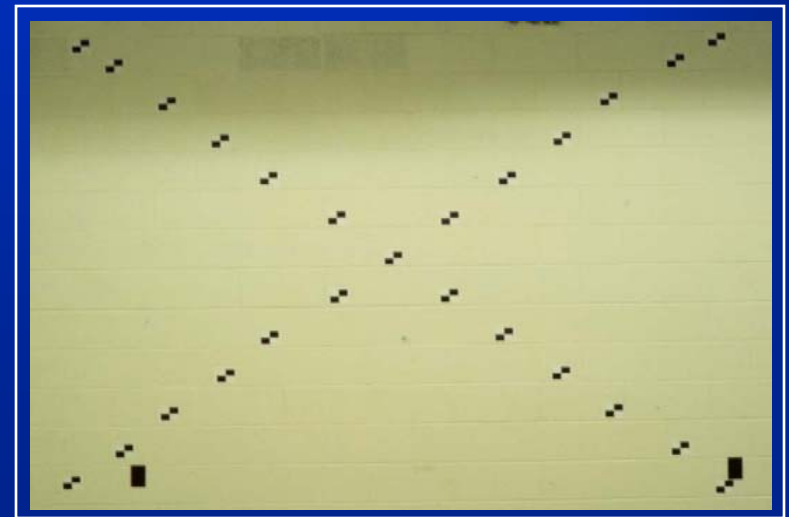
2) Site work

2.1) Calibration site preparation:

- Targets layout
 - Target design
 - cross shape
 - » So their centers positions will be read easily.
 - Targets layout
 - “X” pattern
 - » This layout allows us to recover the needed geometric parameters and systematic errors
- Exposure stations
 - The sensor was mounted on a leveled plate fixed on a survey tripod which is about 8m from targets (far enough to use infinity focus position)



Enlargement showing 5 targets



Targets layout



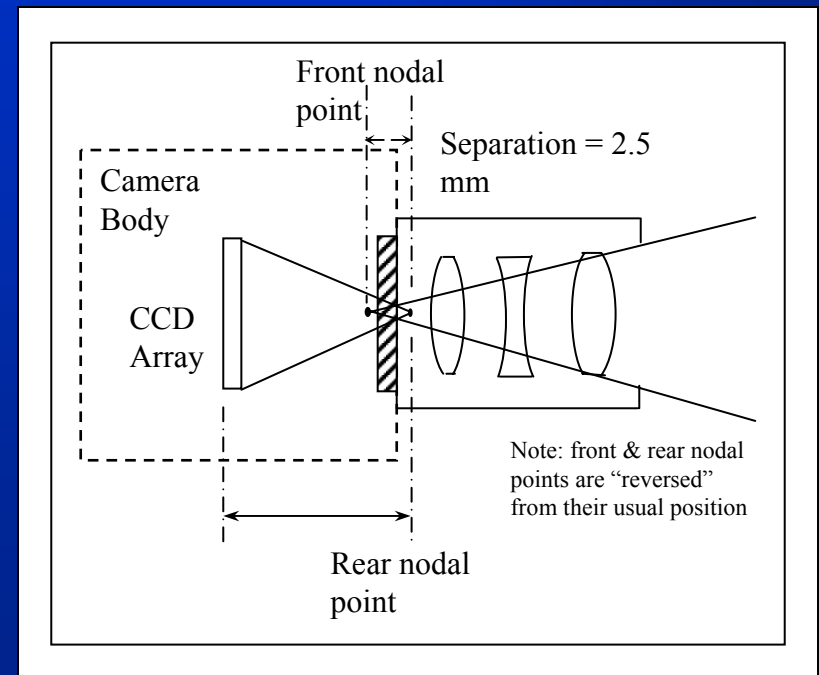
2) Site work steps cont'd

2.2) Measurements and adjustment in object space:

- Laboratory measurements
 - Two 3 arc-second accuracy theodolites were stationed and referenced to measure directions.
 - target centers
 - theodolite locations
 - camera case monuments
 - Steel tape and machinist calipers to measure distances
 - Optical bench to determine the locations of the nodal points



Measurements in the calibration site



cross section of one camera showing the lenses, rear and front nodal points



2) Site work steps cont'd

2.2) Measurements and adjustment in object space cont'd

– Network adjustment

- Having all these observations, we end up with an over determined system of equations.
- A bundle adjustment program was developed to adjust those coordinates of the targets and the instrument stations.
- As a result of that, we determined precisely all our targets and stations in the object space which is a necessity for the calibration purpose.

2.3) Capturing Images

- This step was operated by a team from TEC and observed by the Purdue team
- We tried to simulate the real working conditions by setting the lenses to the “working” infinity focus position
- Images were viewed at the site to make sure that as many as possible of the targets were exposed.



3) Target Locations in Image Space

3.1) Cross Correlation Matching (CCM)

- A cross correlation matching program was used to get rough approximation of target positions in the image space to within a pixel.
- This was done by computing the similarity between a window patch containing the ideal target and the another window from the captured image.
- Despite the fact that, the CCM results showed that we are only away from the exact position by a pixel or less, we needed more accurate and precise methods to define the exact location within a hundredth of a pixel or so.

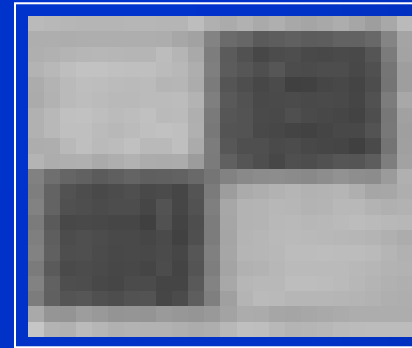
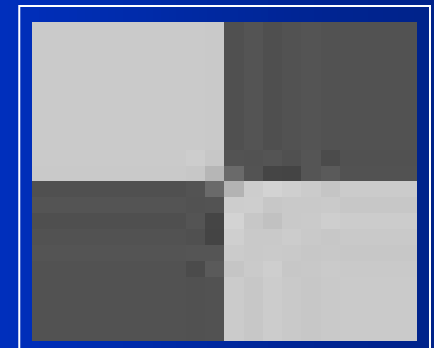
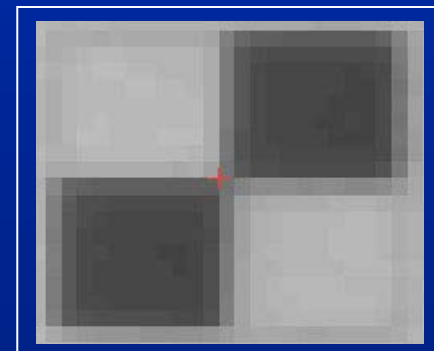


image patch (v)



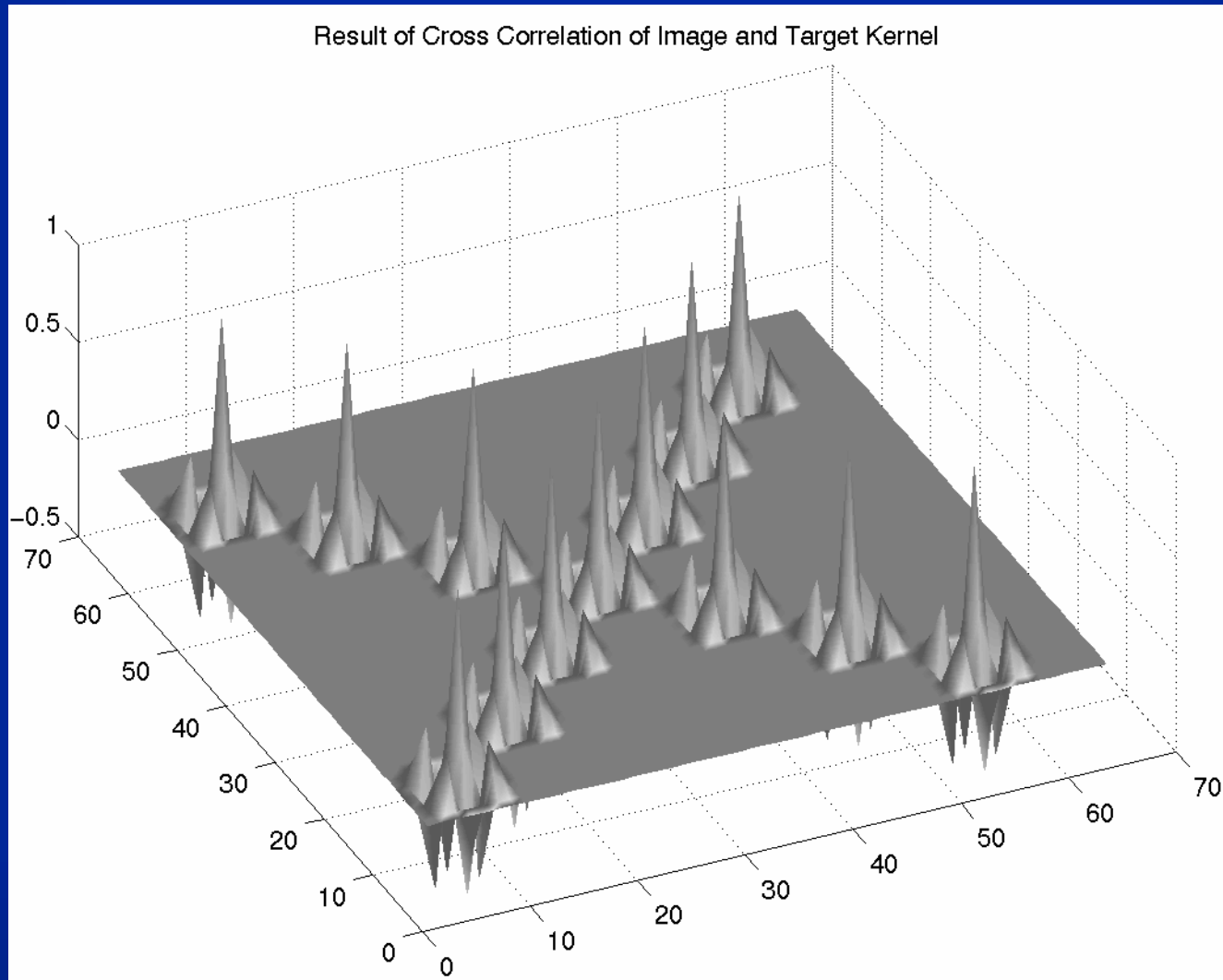
ideal template (u)

$$C_{uv} = \frac{\sum_{i=1}^N (u_i - \bar{u}) (v_i - \bar{v})}{\left[\sum_{i=1}^N (u_i - \bar{u}) \sum_{i=1}^N (v_i - \bar{v}) \right]^{1/2}}$$



CCM Match

Response to Coarse Alignment via Template Matching

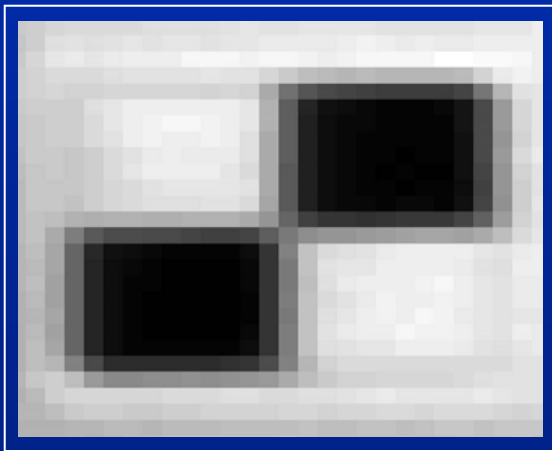




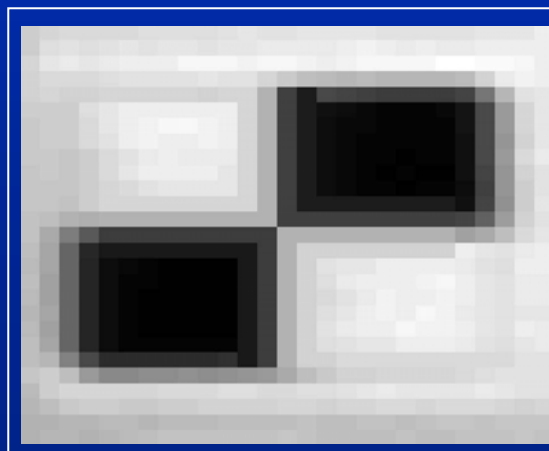
3) Target Locations in Image Space cont'd

3.2) Least squares matching (LSQM)

- LSQM utilizes the first derivative of the intensity in both x and y directions to refine the best correspondence and the exact matching can be reached by moving one window with respect to the other one while minimizing some of squares of differences of intensity values.
- The similarity between the two spots was only geometrically modeled since the radiometric effect was eliminated.
 - Get the approximate location of the imaged target using the first matching approach. Those locations should be within a few pixels of the exact location.
 - A window around that location from the image with adequate size will be extracted.
 - The ideal or template target is retrieved at this point. Similarity in the intensity is enforced between the two windows.



the target in Image patch



the extracted target



matching results using two
methods in succession



4) Math Model and Solution Method

4.1) Math Model

- The mathematical model was chosen carefully in order to recover all significant sources of geometric errors and estimate all significant correction parameters for those errors.

$$x - x_o = x' - x_o + \Delta x = -f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)}$$

$$y - y_o = y' - y_o + \Delta y = -f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)}$$

- Samtaney (1999) explored this model in detail.
 - It was derived from the fundamental collinearity equations. It maps the coordinates from the object space into the image space through some parameters.
 - Exterior parameters which include the location and orientation parameters
 - Interior parameters, lens distortion and focal length are examples of the second type.



4) Math Model cont'd

- The model specifically covers and takes into account the lens distortion through some parameters that model radial, decentering, and affinity distortion.

$$\Delta x = \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2 \bar{x} \bar{y}$$

$$\Delta y = \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 \bar{x} \bar{y} + p_2(r^2 + 2\bar{y}^2) + a_1 \bar{x} + a_2 \bar{y}$$

where: $r^2 = \bar{x}^2 + \bar{y}^2$, $\bar{x} = x' - x_o$, $\bar{y} = y' - y_o$

k_i , p_i , a_i : radial , decentering, affinity distortion coefficients



4) Math Model cont'd

$$F_x = \bar{x} + \bar{x}(k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2\bar{x}^2) + 2p_2\bar{x}\bar{y} \\ + f \frac{r_{11}(X - X_c) + r_{12}(Y - Y_c) + r_{13}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0$$

$$F_y = \bar{y} + \bar{y}(k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1\bar{x}\bar{y} + p_2(r^2 + 2\bar{y}^2) + \\ a_1\bar{x} + a_2\bar{y} + f \frac{r_{21}(X - X_c) + r_{22}(Y - Y_c) + r_{23}(Z - Z_c)}{r_{31}(X - X_c) + r_{32}(Y - Y_c) + r_{33}(Z - Z_c)} = 0$$

- Each target observation will generate two equations. Consequently, the number of equations will be twice the number of targets in the image for each camera.



4) Math Model cont'd

4.2) Solution Method

- In a calibration process like this we want the number of equations to exceed the minimum requirement in order to have an *overdetermined* system. Such a system enhances the reliability and precision of the result. In this problem, 13 parameters were estimated and the number of targets that were used was 20 and 21 in some cases. So, the redundancy we had during the procedure was 27.
- The Unified least squares approach was used to solve this system since some *a priori* knowledge is available for a number of parameters. This *a priori* knowledge is utilized to give those parameters initial values and weights.
- In this sense, some of the parameters were treated as observations with low precision by assigned large variances to them. Since the system is non-linear, the parameter values will be updated iteratively until convergence by adding the correction vector



5) Distortion Analysis

- After determining the camera parameters including distortion parameters, distortion curves were drawn for visual and computational analysis.

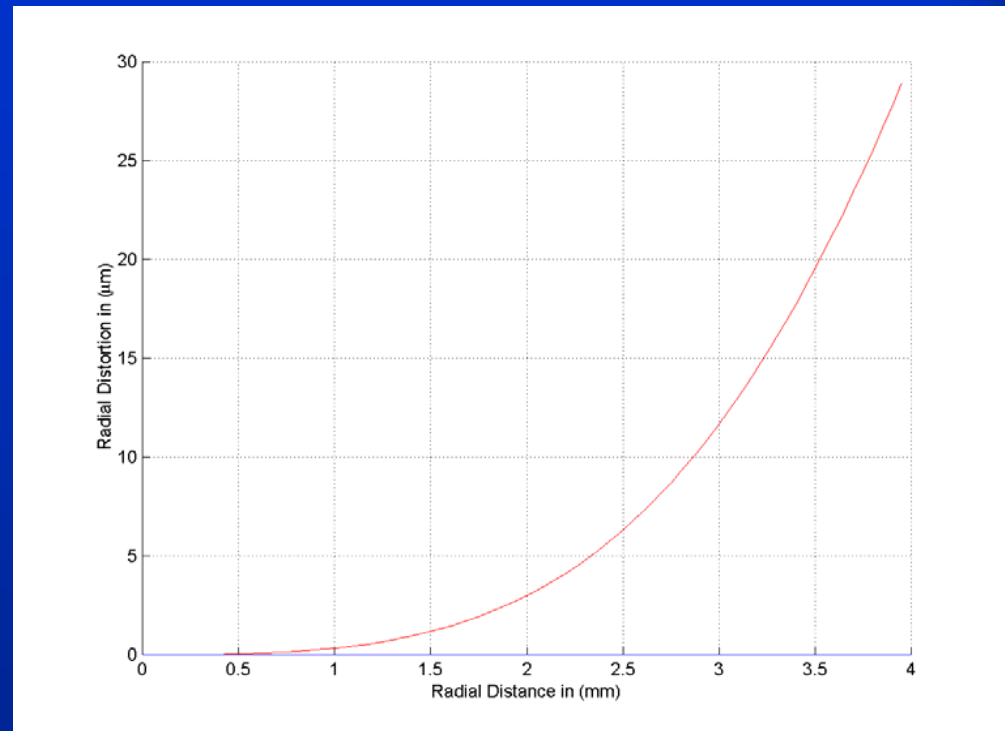
5.1) Radial Distortion

It is the displacement of an imaged object radially either towards or away from the principle point

$$\Delta r = k_1 r^3 + k_2 r^5 + k_3 r^7$$

$$\delta_x = \Delta r * \bar{x} / r \quad \delta_y = \Delta r * \bar{y} / r$$

The resulting curves were obtained for all four cameras and the maximum radial distortion was less than 40 micrometers.



Radial distortion curve of the blue camera

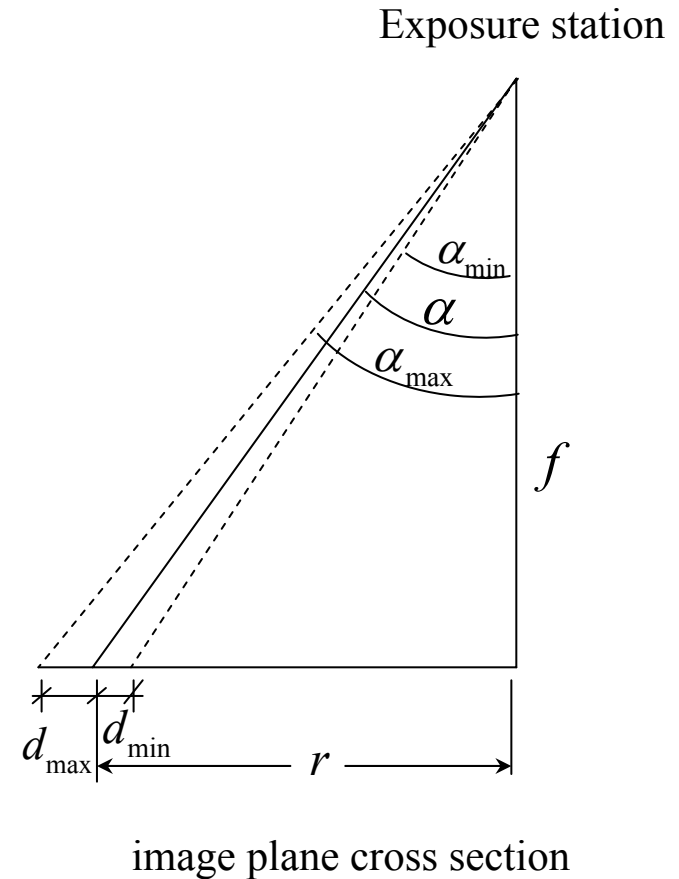


5) Distortion Analysis cont'd

- Radial distortion curve equalization
- The following step was done to level or balance the curve based on equalizing the maximum and the minimum distortion values. This step has no effect on the final results of the corrected coordinates; it is just cosmetic but accepted professional practice.
- Mathematically, balancing the curve leads to a change in the radial distortion parameters and consequently the focal length and other related camera parameters. So this procedure was done iteratively and the parameters were updated.
- CFL: Calibrated focal length.

$$r_{\max} - CFL \times \tan(\alpha_{\max}) + r_{\min} - CFL \times \tan(\alpha_{\min}) = 0$$

$$CFL = \frac{r_{\max} + r_{\min}}{\tan(\alpha_{\max}) + \tan(\alpha_{\min})}$$

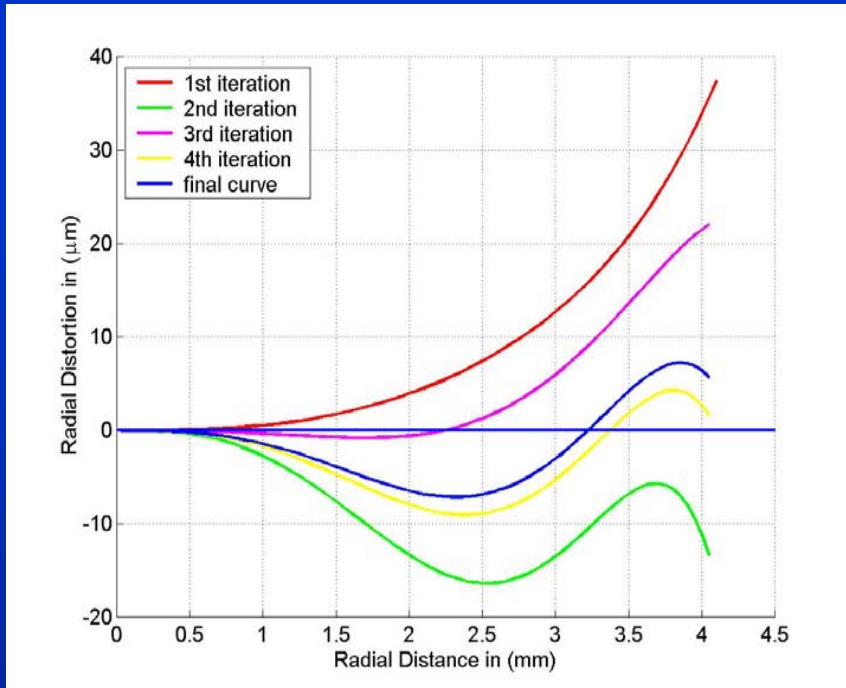


We want d_{\max} and d_{\min} to have approximately the same magnitude.



5) Distortion Analysis cont'd

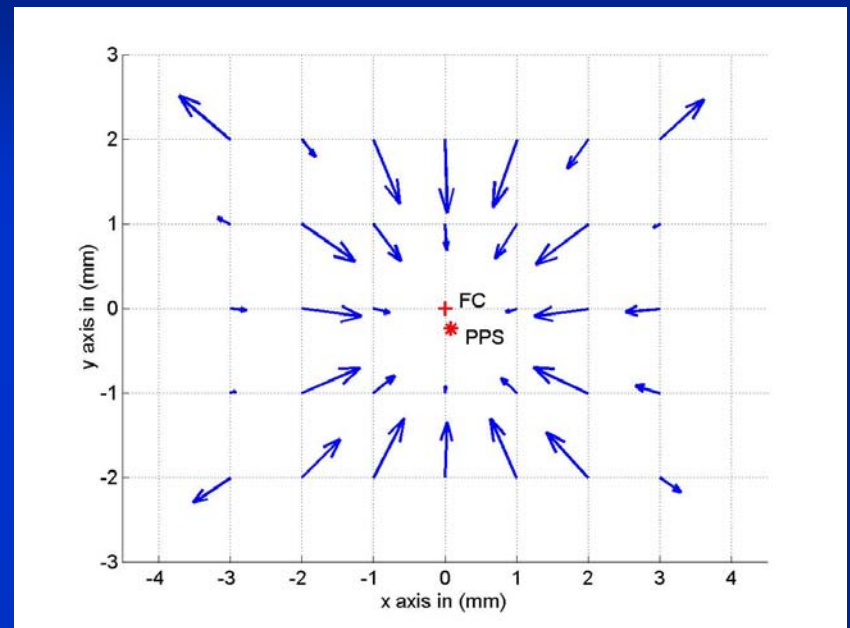
Radial distortion (RD) resulting curves



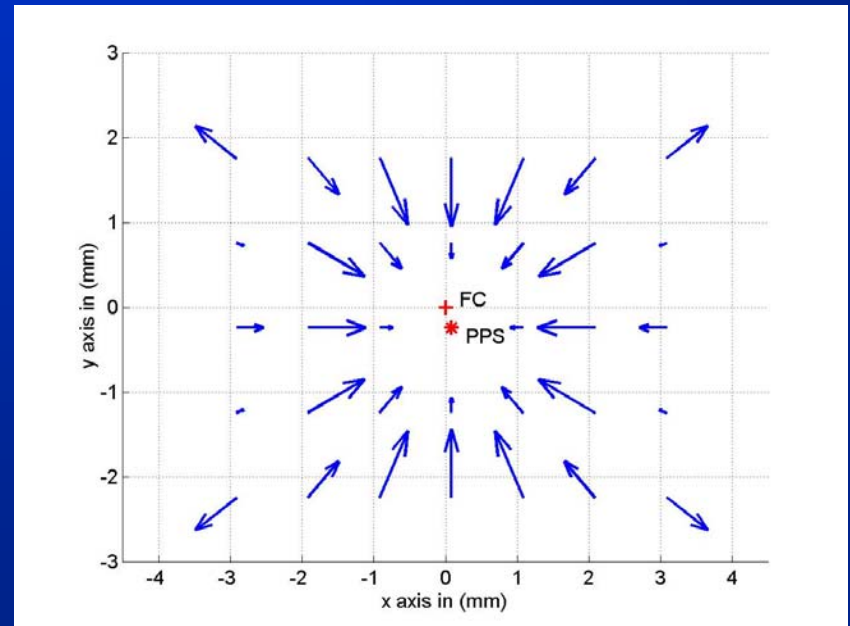
Radial distortion (Equalization iteration
for the Blue camera)

FC: fiducial center.

PPS: principal point of best symmetry.



Scaled RD centered at FC for the Blue camera



Scaled RD centered at PPS for the Blue camera



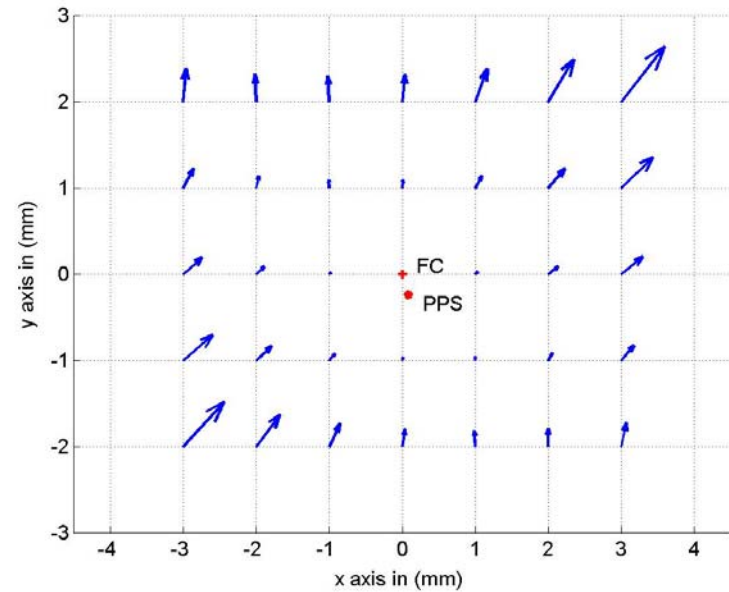
5) Distortion Analysis cont'd

5.2) Decentering Distortion (DD)

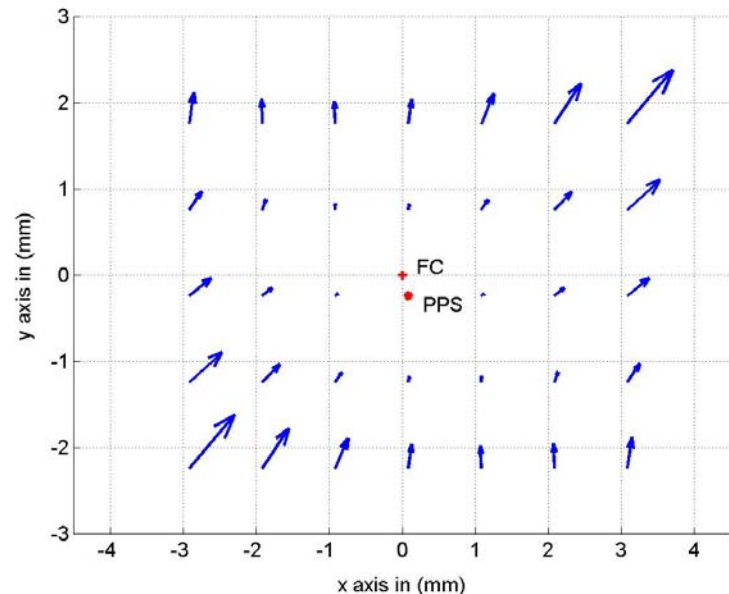
- The misalignment between lens components will lead to systematic image displacement errors which is called Decentering Distortion.

$$\delta_x = p_1[r^2 + 2(x - x_o)^2] + 2p_2(x - x_o)(y - y_o)$$

$$\delta_y = p_2[r^2 + 2(y - y_o)^2] + 2p_1(x - x_o)(y - y_o)$$



Scaled DD centered at PPS for the Blue camera



Scaled DD centered at FC for the Blue camera



6) Results and Discussion

Parameter	Blue Camera (Working band 450nm)	Green Camera (Working band 550nm)	Red Camera (Working band 650nm)	IR Camera (Working band 800nm)
f	16.168 <i>mm</i>	16.154 <i>mm</i>	16.177 <i>mm</i>	16.174 <i>mm</i>
x_o	0.084357 <i>mm</i>	-0.024693 <i>mm</i>	-0.253464 <i>mm</i>	-0.014537 <i>mm</i>
y_o	-0.239466 <i>mm</i>	- 0.213401 <i>mm</i>	- 0.109878 <i>mm</i>	-0.158923 <i>mm</i>
k_1	$-1.696019 * 10^{-3}$	$-1.709735 * 10^{-3}$	$-2.011666 * 10^{-3}$	$-1.712994 * 10^{-3}$
k_2	$0.257710 * 10^{-3}$	$0.274246 * 10^{-3}$	$0.337008 * 10^{-3}$	$0.260639 * 10^{-3}$
k_3	$-0.009089 * 10^{-3}$	$-0.010202 * 10^{-3}$	$-0.013096 * 10^{-3}$	$-0.009190 * 10^{-3}$
p_1	$0.053187 * 10^{-3}$	$0.036416 * 10^{-3}$	$-0.134155 * 10^{-3}$	$-0.009700 * 10^{-3}$
p_2	$0.113262 * 10^{-3}$	$0.036419 * 10^{-3}$	$0.066029 * 10^{-3}$	$0.014206 * 10^{-3}$
a_1	$1.162203 * 10^{-3}$	$0.279185 * 10^{-3}$	$-0.059959 * 10^{-3}$	$0.130802 * 10^{-3}$
a_2	$0.059299 * 10^{-3}$	$-0.663338 * 10^{-3}$	$-0.637141 * 10^{-3}$	$0.278204 * 10^{-3}$

Estimated parameters of the four sensors.



6) Results and discussion cont'd

- For simplicity, the graphical user interface feature in MATLAB was used to create a small user-friendly program to show the results and compute refined image coordinates on a single point basis.
- The parameters of each camera were stored in the file and by inserting the value of the measured line and sample and specifying the corresponding camera, the corrected line and sample will be calculated and printed in the window for the user.

**CAMIS SENSOR
Coordinate Adjustor**

**Input Line and Sample
in pixels**

Line	200
Sample	100

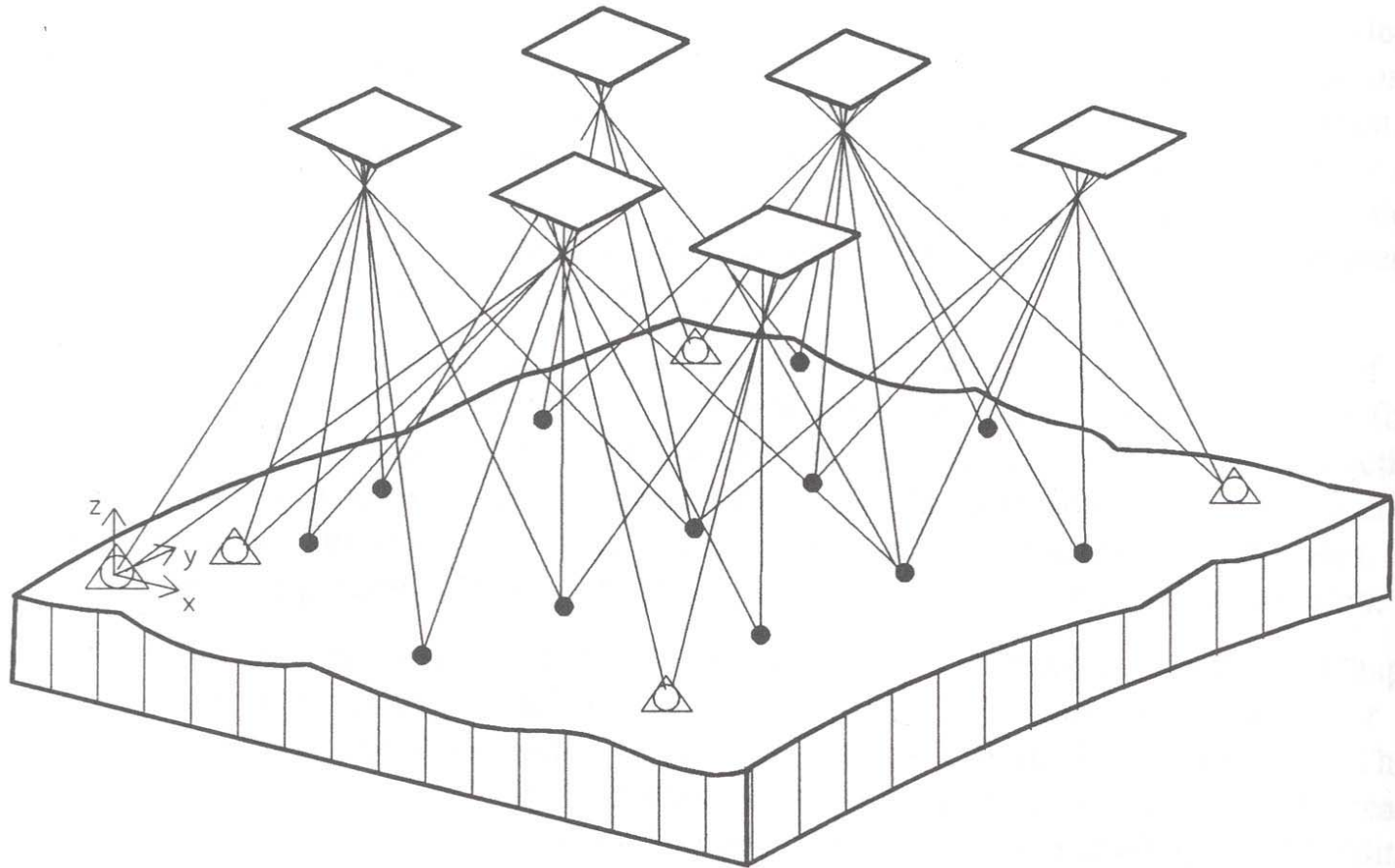
Camera Type

☐ Blue ☒ Green ☐ Red ☐ NIR

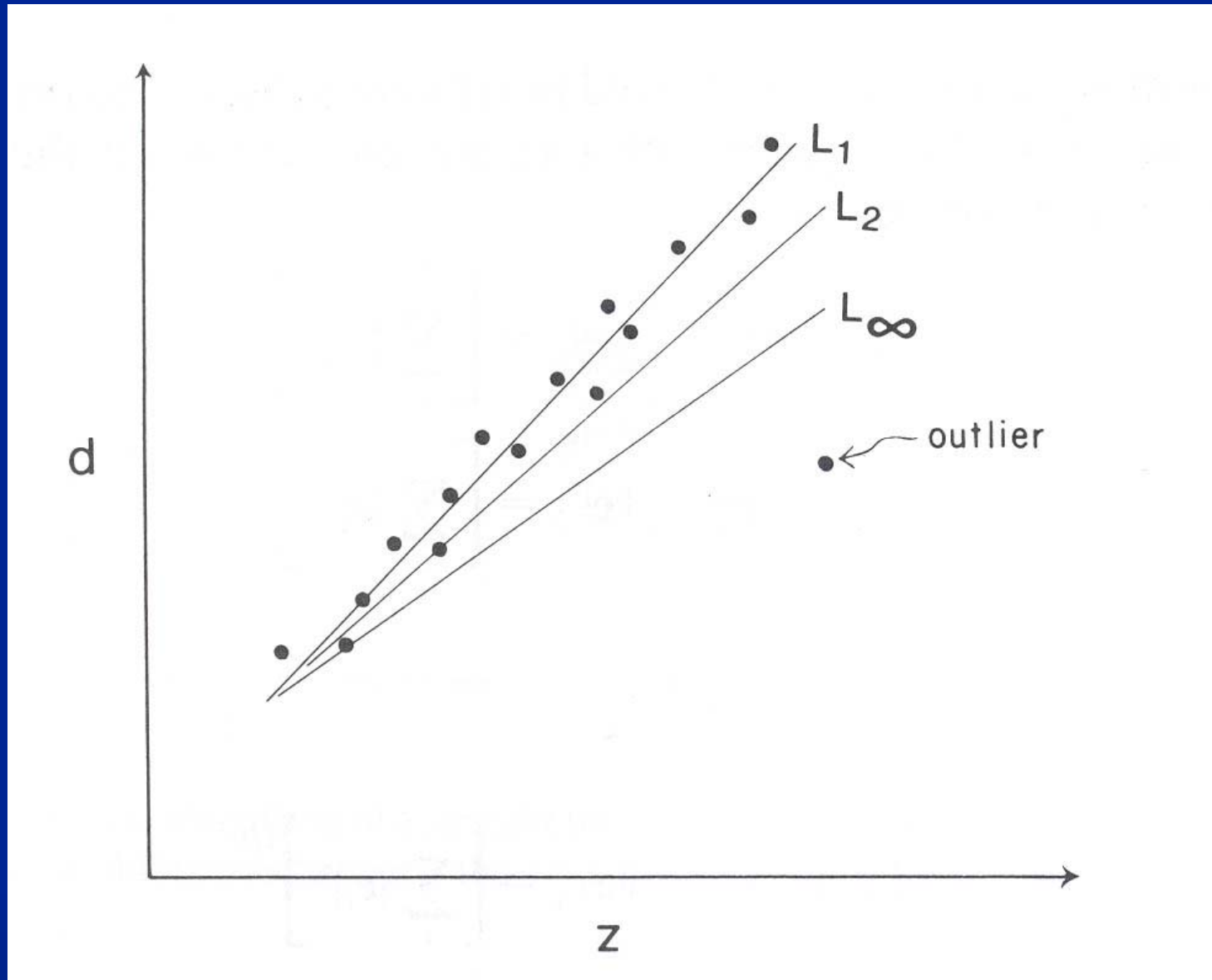
**Corrected Line and
Sample in pixels**

Line	200.4128
Sample	100.7364

Large Blocks May Require Robust Estimation



Possible Techniques: Data Snooping, IRLS, L1, LMS, L-M



Blocks Often Have Regular Structure of Normal Equations

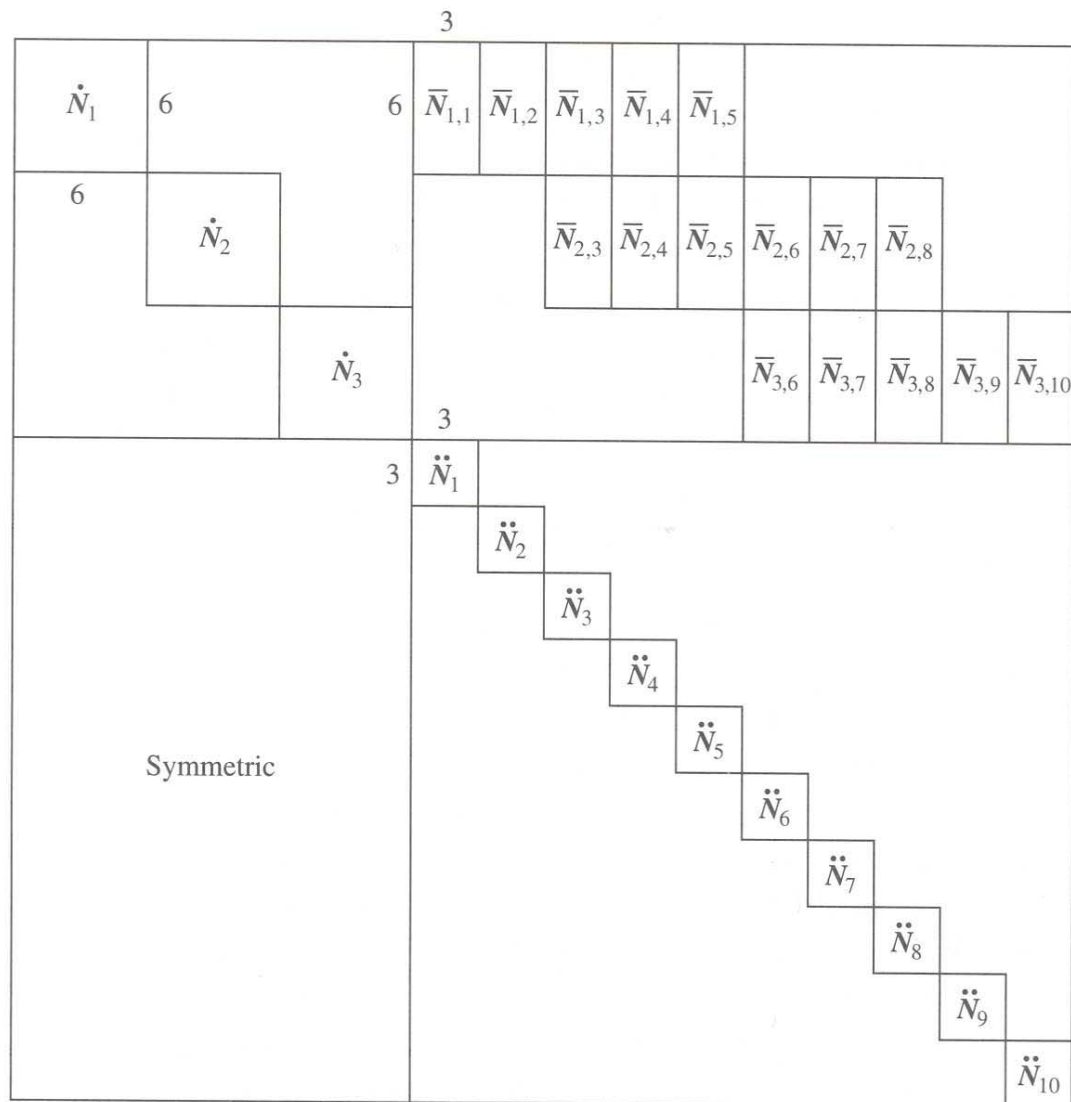


Figure 5-14 Unreduced normal equation structure.

Parameter Ordering with Regular Blocks Creates Sparse Matrix

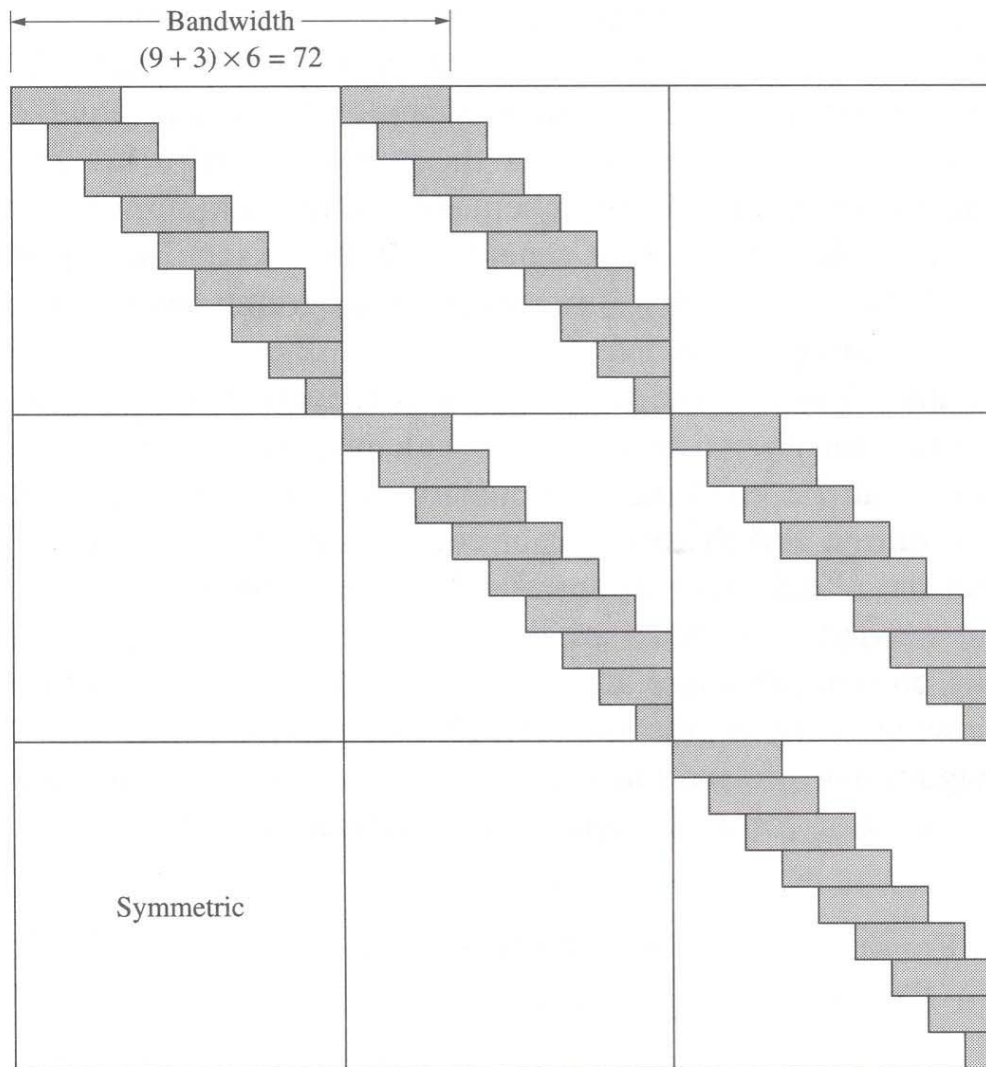


Figure 5-16 Reduced normal equations using down-strip numbering, with three flightlines and nine photos/strip.



The interesting contributions of this research

- We tried during this work to automate the calibration process as much as possible.
- The setup and measurements of the targets and the cameras.
- The automation of the target locations in the images, and their subsequent refinement.
- The automatic process for balancing the radial lens distortion in the presence of other correlated parameters.

- **ACKNOWLEDGEMENT**

- We would like to acknowledge the support of the Army Research Office and the Topographic Engineering Center.

Blunder Detection in Aerial Triangulation

Three algorithms were tested for the purpose of detecting and locating blunders in the triangulation of conventional, frame aerial photographs. It is believed that results from this limited testing will actually apply to a much wider selection of camera and sensor models. A MATLAB coding of the collinearity based bundle block algorithm was produced with a graphical user interface (GUI) for ease of running and evaluating. The data selected for the testing was a block of two images from a larger clock of about 30 images taken over the Purdue University campus. The scale is nominally 1:4000, the two photos selected 1-7 and 1-9 were a 60% overlap pair. The block actually contained 80% forward overlap imagery but the intervening image(1-8) was omitted to reflect the more common collection geometry.

The photos are shown below.



Figure 1 Left Photo 1-7



Figure 2. Right Photo 1-9

50 points have been read, distributed over the stereo model, including 3 ground control points. Errors have been artificially inserted into 3 of the observations to evaluate the three different adjustment algorithms. (1) L2 norm minimization (least squares) converges and it is clear that errors are present, but it is difficult to locate them because the effects are spread around neighboring points. (2) likewise IRLS (iteratively reweighted least squares) works well, but the weight function for adjusting the weights at each iteration seems disturbingly dependent on the particular application. (3) L1 norm minimization appears to work the best. As one can see from the numerical results and the graphical depiction of the residual vectors, all of the error is correctly placed into the 3 observations where it originally occurred. They were all y-errors since, in a 2 model block, x-errors are not detectable.

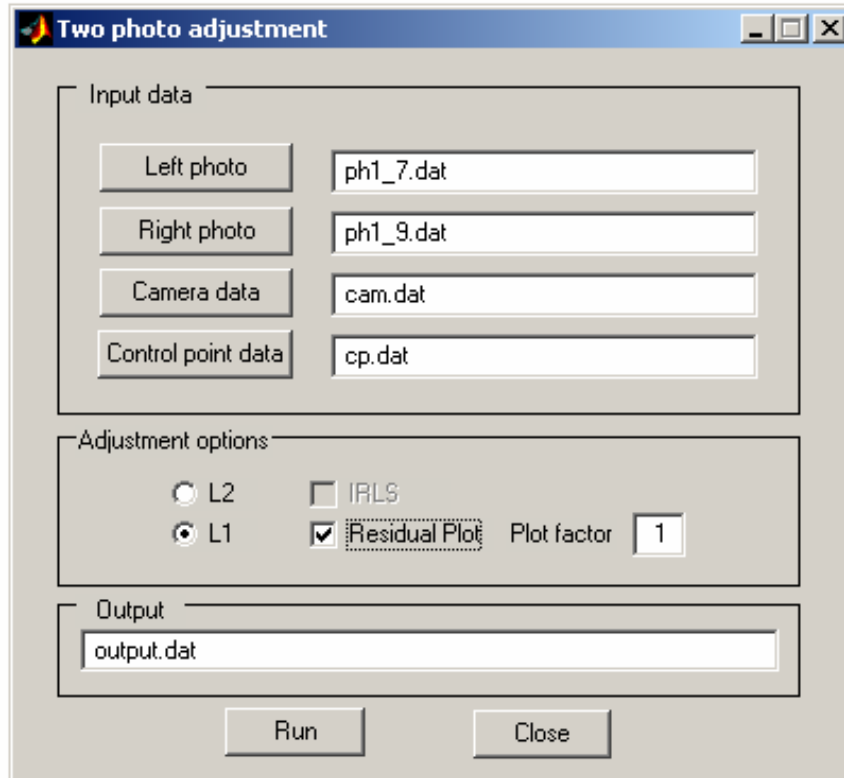


Figure 3. GUI Program with selections for L1

The output listing showing the 5 iterations and the final residuals follows.

```
Number of control points: 3
Number of total parameters: 162
Number of adjusted parameters:155
Number of equations: 200
Number of fixed parameters: 7
Redundancy: 45
```

```
begin iterations
```

```
*****
```

```
iter =
```

```
1
```

```
Optimization terminated successfully.
```

```
parameter corrections
```

```
pc =
```

```
-0.0085  -0.0024  -0.0009  -3.9883   5.0264  -1.3021
 0.0374  -0.0366  -0.0509  -2.7118   4.8994   1.6004
```

```
dspv =
```

```
1.0000   0.0000   0.4914   0.0000  -0.0000
2.0000   0.0000   0.7580   0.0000  -0.0000
3.0000   0.0000   0.1657   0.0000  -0.0000
101.0000  0.0000  30.6690   0.0000  -0.0000
102.0000  0.0000  -0.0072   0.0000  -0.0000
103.0000  0.0000   0.0122   0.0000  -0.0000
104.0000  0.0000  -0.0000   0.0000  -0.0000
```

105.0000	0.0000	30.1345	0.0000	-0.0000
106.0000	0.0000	-0.0464	0.0000	-0.0000
107.0000	0.0000	-0.0245	0.0000	-0.0000
108.0000	0.0000	-0.0099	0.0000	-0.0000
109.0000	0.0000	0.0380	0.0000	-0.0000
110.0000	0.0000	-0.0376	0.0000	-0.0000
111.0000	0.0000	0.0000	0.0000	-0.0000
112.0000	0.0000	0.0256	0.0000	-0.0000
113.0000	0.0000	0.0363	0.0000	-0.0000
114.0000	0.0000	-0.0148	0.0000	-0.0000
115.0000	0.0000	-0.0454	0.0000	-0.0000
116.0000	0.0000	-0.0300	0.0000	-0.0000
117.0000	0.0000	-0.0241	0.0000	-0.0000
118.0000	0.0000	0.0552	0.0000	-0.0000
119.0000	0.0000	0.0053	0.0000	-0.0000
120.0000	0.0000	-0.0586	0.0000	-0.0000
121.0000	0.0000	-0.0311	-0.0000	-0.0000
122.0000	0.0000	0.0187	-0.0000	-0.0000
123.0000	0.0000	-0.0146	0.0000	-0.0000
124.0000	0.0000	-0.0268	-0.0000	-0.0000
125.0000	0.0000	28.9544	0.0000	-0.0000
201.0000	0.0000	-0.0153	0.0000	-0.0000
202.0000	0.0000	0.0288	0.0000	-0.0000
203.0000	0.0000	0.0107	0.0000	-0.0000
204.0000	0.0000	-0.0205	0.0000	-0.0000
205.0000	0.0000	0.0065	0.0000	-0.0000
206.0000	0.0000	-0.0317	0.0000	-0.0000
207.0000	0.0000	0.0256	0.0000	-0.0000
208.0000	0.0000	0.0000	-0.0000	-0.0000
209.0000	0.0000	0.0047	-0.0000	-0.0000
210.0000	0.0000	-0.0016	0.0000	-0.0000
211.0000	0.0000	0.0137	0.0000	-0.0000
212.0000	0.0000	-0.0000	0.0000	-0.0000
213.0000	0.0000	-0.0623	0.0000	-0.0000
214.0000	0.0000	0.0182	0.0000	-0.0000
215.0000	0.0000	0.0187	-0.0000	-0.0000
218.0000	0.0000	0.0087	0.0000	-0.0000
219.0000	0.0000	-0.0000	0.0000	-0.0000
220.0000	0.0000	-0.0087	0.0000	-0.0000
221.0000	0.0000	-0.0416	0.0000	-0.0000
222.0000	0.0000	-0.0507	0.0000	-0.0000
223.0000	0.0000	-0.0036	0.0000	-0.0000
225.0000	0.0000	-0.0388	0.0000	-0.0000

iter =

2

Optimization terminated successfully.

parameter corrections

pc =

0.0080	0.0018	0.0010	3.5852	-4.7558	1.3144
0.0063	-0.0002	0.0022	2.7585	-4.5207	-1.2844

dspv =

1.0000	-0.0000	0.0000	-0.0000	-0.0000
2.0000	-0.0000	0.0000	-0.0000	-0.0099
3.0000	-0.0000	-0.0000	0.0000	-0.0000
101.0000	-0.0000	29.9364	-0.0000	-0.0000

102.0000	-0.0000	-0.0185	-0.0000	-0.0000
103.0000	-0.0000	0.0060	-0.0000	-0.0000
104.0000	-0.0000	0.0017	-0.0000	-0.0000
105.0000	-0.0000	29.9776	-0.0000	-0.0000
106.0000	-0.0000	-0.0451	-0.0000	-0.0000
107.0000	-0.0000	-0.0061	-0.0000	-0.0000
108.0000	-0.0000	-0.0086	-0.0000	-0.0000
109.0000	-0.0000	0.0390	-0.0000	-0.0000
110.0000	-0.0000	0.0000	-0.0000	0.0138
111.0000	-0.0000	0.0105	-0.0000	-0.0000
112.0000	-0.0000	0.0000	-0.0000	-0.0280
113.0000	-0.0000	0.0000	-0.0000	-0.0362
114.0000	-0.0000	0.0000	-0.0000	0.0057
115.0000	-0.0000	0.0000	-0.0000	0.0161
116.0000	-0.0000	-0.0000	-0.0000	0.0207
117.0000	-0.0000	-0.0000	-0.0000	0.0262
118.0000	0.0000	0.0000	-0.0000	-0.0538
119.0000	-0.0000	0.0000	0.0000	-0.0058
120.0000	-0.0000	0.0000	-0.0000	0.0359
121.0000	-0.0000	-0.0000	-0.0000	0.0313
122.0000	-0.0000	0.0000	-0.0000	-0.0000
123.0000	-0.0000	0.0000	-0.0000	0.0275
124.0000	-0.0000	0.0000	-0.0000	0.0109
125.0000	-0.0000	0.0000	-0.0000	-28.4591
201.0000	-0.0000	-0.0267	-0.0000	-0.0000
202.0000	-0.0000	0.0342	-0.0000	-0.0000
203.0000	-0.0000	0.0041	-0.0000	-0.0000
204.0000	-0.0000	-0.0091	-0.0000	-0.0000
205.0000	-0.0000	0.0035	-0.0000	-0.0000
206.0000	-0.0000	-0.0304	-0.0000	-0.0000
207.0000	0.0000	0.0358	-0.0000	-0.0000
208.0000	-0.0000	0.0000	-0.0000	-0.0000
209.0000	-0.0000	-0.0000	-0.0000	0.0103
210.0000	-0.0000	0.0000	-0.0000	0.0187
211.0000	-0.0000	0.0000	-0.0000	-0.0050
212.0000	-0.0000	0.0000	-0.0000	-0.0055
213.0000	-0.0000	0.0000	-0.0000	0.0447
214.0000	-0.0000	0.0000	-0.0000	-0.0179
215.0000	0.0000	0.0000	-0.0000	-0.0190
218.0000	-0.0000	0.0000	-0.0000	-0.0000
219.0000	-0.0000	0.0000	-0.0000	-0.0284
220.0000	-0.0000	-0.0032	-0.0000	-0.0000
221.0000	-0.0000	-0.0000	-0.0000	0.0150
222.0000	0.0000	-0.0561	-0.0000	-0.0000
223.0000	-0.0000	-0.0032	-0.0000	-0.0000
225.0000	-0.0000	0.0000	-0.0000	0.0343

iter =

3

Optimization terminated successfully.

parameter corrections

pc =

-0.0001	-0.0000	-0.0000	-0.0108	0.0651	-0.0398
0.0000	-0.0001	-0.0000	-0.0712	-0.0450	0.0085

dspv =

1.0000	0.0000	-0.0007	0.0000	0.0000
--------	--------	---------	--------	--------

2.0000	-0.0000	0.0000	-0.0000	-0.0164
3.0000	0.0000	0.0000	0.0000	0.0000
101.0000	0.0000	29.9454	0.0000	0.0000
102.0000	0.0000	-0.0128	0.0000	0.0000
103.0000	0.0000	0.0113	0.0000	0.0000
104.0000	0.0000	0.0036	0.0000	0.0000
105.0000	0.0000	29.9858	0.0000	0.0000
106.0000	0.0000	-0.0427	0.0000	0.0000
107.0000	0.0000	-0.0067	0.0000	0.0000
108.0000	0.0000	-0.0084	0.0000	0.0000
109.0000	0.0000	0.0400	0.0000	0.0000
110.0000	0.0000	-0.0000	0.0000	0.0112
111.0000	0.0000	0.0091	0.0000	0.0000
112.0000	-0.0000	0.0000	-0.0000	-0.0276
113.0000	-0.0000	0.0000	-0.0000	-0.0375
114.0000	-0.0000	0.0000	0.0000	0.0019
115.0000	-0.0000	0.0000	0.0000	0.0098
116.0000	-0.0000	0.0000	0.0000	0.0224
117.0000	-0.0000	0.0000	-0.0000	0.0249
118.0000	-0.0000	0.0000	-0.0000	-0.0563
119.0000	-0.0000	0.0000	-0.0000	-0.0101
120.0000	-0.0000	0.0000	-0.0000	0.0275
121.0000	-0.0000	0.0000	-0.0000	0.0312
122.0000	-0.0000	0.0000	-0.0000	-0.0061
123.0000	-0.0000	0.0000	-0.0000	0.0181
124.0000	-0.0000	0.0000	-0.0000	0.0000
125.0000	-0.0000	0.0000	0.0000	-28.6231
201.0000	0.0000	-0.0199	0.0000	0.0000
202.0000	0.0000	0.0381	0.0000	0.0000
203.0000	0.0000	0.0066	0.0000	0.0000
204.0000	0.0000	-0.0081	0.0000	0.0000
205.0000	0.0000	0.0040	0.0000	0.0000
206.0000	0.0000	-0.0319	0.0000	0.0000
207.0000	0.0000	0.0000	-0.0000	-0.0333
208.0000	-0.0000	0.0000	-0.0000	0.0000
209.0000	-0.0000	0.0000	-0.0000	0.0083
210.0000	-0.0000	0.0000	-0.0000	0.0154
211.0000	-0.0000	0.0000	-0.0000	-0.0140
212.0000	-0.0000	0.0000	-0.0000	-0.0204
213.0000	-0.0000	0.0000	-0.0000	0.0357
214.0000	-0.0000	0.0000	-0.0000	-0.0189
215.0000	-0.0000	0.0000	-0.0000	-0.0201
218.0000	0.0000	0.0000	0.0000	0.0000
219.0000	0.0000	0.0000	0.0000	-0.0316
220.0000	0.0000	-0.0017	0.0000	0.0000
221.0000	0.0000	-0.0000	0.0000	0.0122
222.0000	0.0000	-0.0515	0.0000	0.0000
223.0000	0.0000	0.0000	0.0000	0.0000
225.0000	-0.0000	0.0000	-0.0000	0.0261

iter =

4

Optimization terminated successfully.

parameter corrections

pc =

1.0e-004 *

-0.0004	0.0002	-0.0001	0.2307	0.1752	-0.2362
0.0001	0.0002	0.0001	0.1171	-0.0445	-0.1519
dspv =					
1.0000	0.0000	-0.0007	0.0000	0.0000	
2.0000	-0.0000	-0.0000	0.0000	-0.0164	
3.0000	0.0000	-0.0000	0.0000	-0.0000	
101.0000	0.0000	29.9454	0.0000	-0.0000	
102.0000	0.0000	-0.0128	0.0000	-0.0000	
103.0000	0.0000	0.0113	0.0000	0.0000	
104.0000	0.0000	0.0036	0.0000	0.0000	
105.0000	0.0000	29.9858	0.0000	0.0000	
106.0000	0.0000	-0.0427	0.0000	-0.0000	
107.0000	0.0000	-0.0067	0.0000	-0.0000	
108.0000	0.0000	-0.0084	0.0000	0.0000	
109.0000	0.0000	0.0400	0.0000	-0.0000	
110.0000	0.0000	-0.0000	0.0000	0.0112	
111.0000	-0.0000	0.0091	0.0000	-0.0000	
112.0000	-0.0000	0.0000	0.0000	-0.0276	
113.0000	-0.0000	0.0000	0.0000	-0.0375	
114.0000	-0.0000	-0.0000	0.0000	0.0019	
115.0000	-0.0000	-0.0000	0.0000	0.0098	
116.0000	-0.0000	-0.0000	0.0000	0.0224	
117.0000	-0.0000	-0.0000	0.0000	0.0249	
118.0000	-0.0000	0.0000	0.0000	-0.0563	
119.0000	-0.0000	0.0000	0.0000	-0.0101	
120.0000	-0.0000	-0.0000	0.0000	0.0275	
121.0000	-0.0000	-0.0000	0.0000	0.0312	
122.0000	-0.0000	-0.0000	0.0000	-0.0061	
123.0000	-0.0000	-0.0000	0.0000	0.0181	
124.0000	-0.0000	-0.0000	0.0000	0.0000	
125.0000	-0.0000	-0.0000	0.0000	-28.6231	
201.0000	0.0000	-0.0199	-0.0000	-0.0000	
202.0000	0.0000	0.0381	0.0000	-0.0000	
203.0000	0.0000	0.0066	0.0000	-0.0000	
204.0000	0.0000	-0.0081	0.0000	-0.0000	
205.0000	0.0000	0.0040	0.0000	-0.0000	
206.0000	0.0000	-0.0319	0.0000	-0.0000	
207.0000	-0.0000	0.0000	0.0000	-0.0333	
208.0000	-0.0000	-0.0000	0.0000	-0.0000	
209.0000	-0.0000	-0.0000	0.0000	0.0082	
210.0000	-0.0000	0.0000	0.0000	0.0154	
211.0000	-0.0000	-0.0000	0.0000	-0.0140	
212.0000	-0.0000	-0.0000	0.0000	-0.0204	
213.0000	-0.0000	-0.0000	0.0000	0.0357	
214.0000	-0.0000	-0.0000	0.0000	-0.0189	
215.0000	-0.0000	-0.0000	0.0000	-0.0201	
218.0000	0.0000	-0.0000	0.0000	0.0000	
219.0000	0.0000	-0.0000	0.0000	-0.0315	
220.0000	0.0000	-0.0017	0.0000	0.0000	
221.0000	0.0000	-0.0000	0.0000	0.0122	
222.0000	0.0000	-0.0515	0.0000	0.0000	
223.0000	0.0000	-0.0000	0.0000	0.0000	
225.0000	-0.0000	-0.0000	0.0000	0.0261	

we have converged

residuals

point-id	vx-l	vy-l	vx-r	vy-r
# 1	0.00000	-0.00070	0.00000	0.00000
# 2	-0.00000	-0.00000	0.00000	-0.01644
# 3	0.00000	-0.00000	0.00000	-0.00000
# 101	0.00000	29.94538	0.00000	-0.00000
# 102	0.00000	-0.01281	0.00000	-0.00000
# 103	0.00000	0.01127	0.00000	0.00000
# 104	0.00000	0.00361	0.00000	0.00000
# 105	0.00000	29.98575	0.00000	0.00000
# 106	0.00000	-0.04273	0.00000	-0.00000
# 107	0.00000	-0.00669	0.00000	-0.00000
# 108	0.00000	-0.00840	0.00000	0.00000
# 109	0.00000	0.04005	0.00000	-0.00000
# 110	0.00000	-0.00000	0.00000	0.01124
# 111	-0.00000	0.00914	0.00000	-0.00000
# 112	-0.00000	0.00000	0.00000	-0.02765
# 113	-0.00000	0.00000	0.00000	-0.03753
# 114	-0.00000	-0.00000	0.00000	0.00189
# 115	-0.00000	-0.00000	0.00000	0.00980
# 116	-0.00000	-0.00000	0.00000	0.02236
# 117	-0.00000	-0.00000	0.00000	0.02493
# 118	-0.00000	0.00000	0.00000	-0.05632
# 119	-0.00000	0.00000	0.00000	-0.01009
# 120	-0.00000	-0.00000	0.00000	0.02751
# 121	-0.00000	-0.00000	0.00000	0.03116
# 122	-0.00000	-0.00000	0.00000	-0.00609
# 123	-0.00000	-0.00000	0.00000	0.01809
# 124	-0.00000	-0.00000	0.00000	0.00000
# 125	-0.00000	-0.00000	0.00000	-28.62311
# 201	0.00000	-0.01987	-0.00000	-0.00000
# 202	0.00000	0.03806	0.00000	-0.00000
# 203	0.00000	0.00661	0.00000	-0.00000
# 204	0.00000	-0.00810	0.00000	-0.00000
# 205	0.00000	0.00398	0.00000	-0.00000
# 206	0.00000	-0.03191	0.00000	-0.00000
# 207	-0.00000	0.00000	0.00000	-0.03331
# 208	-0.00000	-0.00000	0.00000	-0.00000
# 209	-0.00000	-0.00000	0.00000	0.00825
# 210	-0.00000	0.00000	0.00000	0.01540
# 211	-0.00000	-0.00000	0.00000	-0.01396
# 212	-0.00000	-0.00000	0.00000	-0.02035
# 213	-0.00000	-0.00000	0.00000	0.03568
# 214	-0.00000	-0.00000	0.00000	-0.01888
# 215	-0.00000	-0.00000	0.00000	-0.02013
# 218	0.00000	-0.00000	0.00000	0.00000
# 219	0.00000	-0.00000	0.00000	-0.03155
# 220	0.00000	-0.00175	0.00000	0.00000
# 221	0.00000	-0.00000	0.00000	0.01222
# 222	0.00000	-0.05155	0.00000	0.00000
# 223	0.00000	-0.00000	0.00000	0.00000
# 225	-0.00000	-0.00000	0.00000	0.02608

left photo w,p,k xl,yl,zl

omega_l =

```

-0.0299735226753904
phi_l =
0.0195274155794779
kappa_l =
1.55932710861381
xl_l =
913722.159060762
yl_l =
575234.888671137
zl_l =
758.679524164434
right photo w,p,k xl,yl,zl

omega_r =
0.0143402128413725
phi_r =
-0.0167326674115072
kappa_r =
1.51038139606387
xl_r =
913729.425538108
yl_r =
574874.372672258
zl_r =
761.308438653024
object points x,y,z
1 913565.589 575312.953 188.413
2 913928.656 575198.483 189.867
3 913421.163 574863.304 188.864
101 913449.026 574846.372 188.528
102 913440.177 574922.010 200.675
103 913435.540 574965.220 189.324
104 913518.165 575132.684 188.807
105 913434.248 575308.938 188.547
106 913562.052 574847.230 205.327
107 913697.162 574825.802 191.221
108 913676.814 574975.794 189.464
109 913654.151 575150.062 202.171
110 913613.517 575270.275 189.052
111 913775.073 574879.419 190.584
112 913868.727 574941.724 190.123
113 913868.030 575021.060 190.213
114 913854.092 575141.748 189.913
115 913837.288 575260.592 199.936
116 913942.714 574853.346 189.890
117 913920.787 574981.803 190.062
118 913921.631 575034.847 190.198
119 913941.089 575081.310 189.899
120 913940.164 575218.731 189.726
121 914039.235 574865.686 188.286
122 914108.997 574983.294 188.747
123 914097.685 575078.001 189.460
124 914022.784 575203.542 189.358
125 913919.546 575295.806 195.643
201 913408.619 574931.351 197.214
202 913537.804 574813.597 202.504
203 913537.157 574914.501 202.654

```


204	913619.271	574814.277	202.756
205	913618.728	574915.197	202.532
206	913728.431	574874.684	208.342
207	913885.559	574842.455	199.319
208	914039.119	574865.807	188.294
209	914049.479	574923.479	202.069
210	914050.830	574958.793	201.987
211	914075.889	575091.662	189.508
212	914113.320	575200.907	189.548
213	913963.994	575211.224	200.330
214	913873.859	575002.922	190.056
215	913847.850	575026.485	190.334
218	913673.822	575036.970	206.215
219	913644.580	575282.572	189.009
220	913624.477	575192.311	201.137
221	913616.112	575283.177	188.907
222	913437.545	575004.464	188.946
223	913433.740	575137.764	188.617
225	913993.507	575155.644	200.014

As can be seen by the last residual listing, the three 30mm errors stand out clearly. These are also shown on the following graphical residual plot.

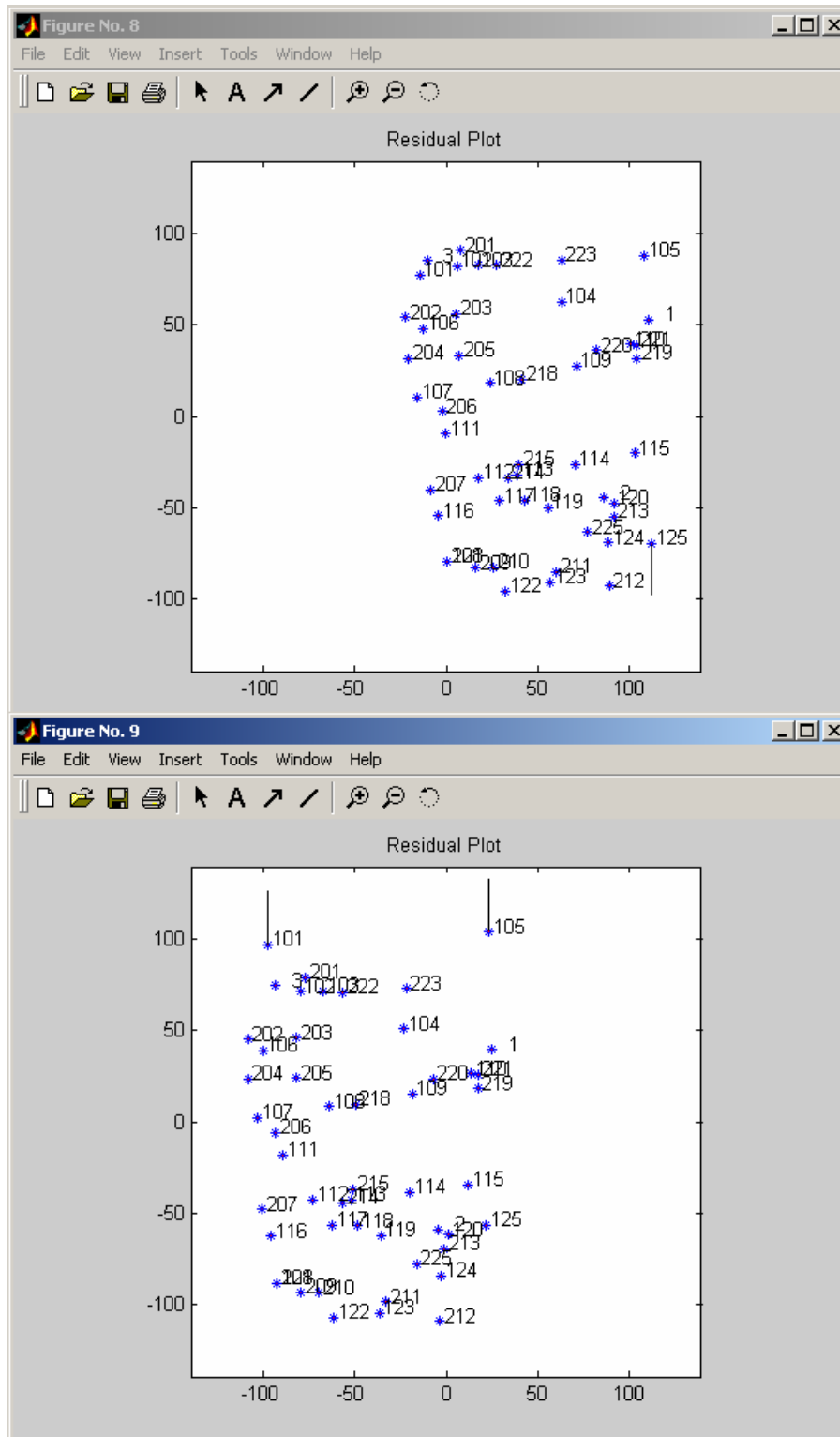


Figure 4. Residual Vectors with 3 Blunders Clearly Evident

Our conclusion from this exercise is that the L1 minimization procedure is a very powerful tool for blunder detection, when there is sufficient redundancy. It is slower than

least squares by a factor of 10-20, but when looking for difficult to find blunders, that time can easily be accommodated versus, manual inspection or univariate data snooping.



BAE SYSTEMS

3D Visualization

Joseph Spann
BAE SYSTEMS
(858)592-5853

September 5, 2001

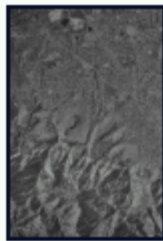
Photogrammetry Using 3-D Graphics and Projective Textures

Joseph R. Spann and Karen S. Kaufman

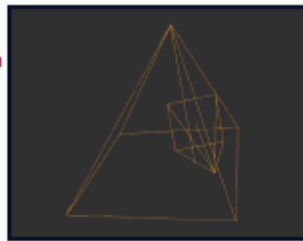
BAE SYSTEMS Mission Solutions



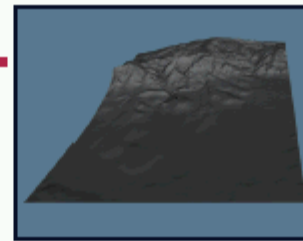
Oblique photograph of
Boulder, Colorado



Aerial photograph of
Boulder, Colorado

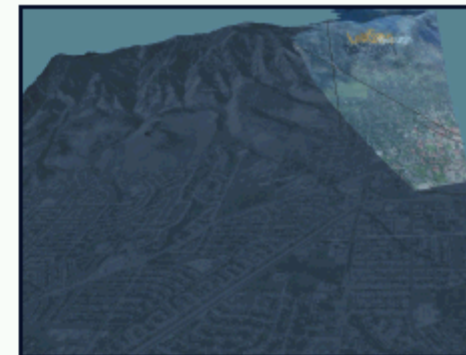


Sensor models



Terrain/feature model of
Boulder, Colorado

Sensor projector combines rigorous photogrammetric camera models with 3-D virtual reality



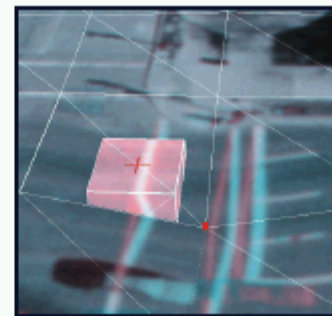
Multiple images simultaneously projected
on terrain and features



Imagery projected on terrain/feature
models, Washington, D.C.



Orthographic view yields true
orthographic projection



Movable projection surface enables
photogrammetric measurements

3D Visualization Background

- VESPER image projection is implemented via a 4 x 4 projection matrix.
- For a frame camera, the projection matrix elements are derived from the frame camera parameters.
- For an orthographic projection, they are readily derived from the orthographic projection parameters.
- Complex sensor models cannot be converted into a single 4 x 4 projection matrix.
- VESPER is being extended to accurately project whole images from sensors other than frame or orthographic.

3D Visualization Objectives

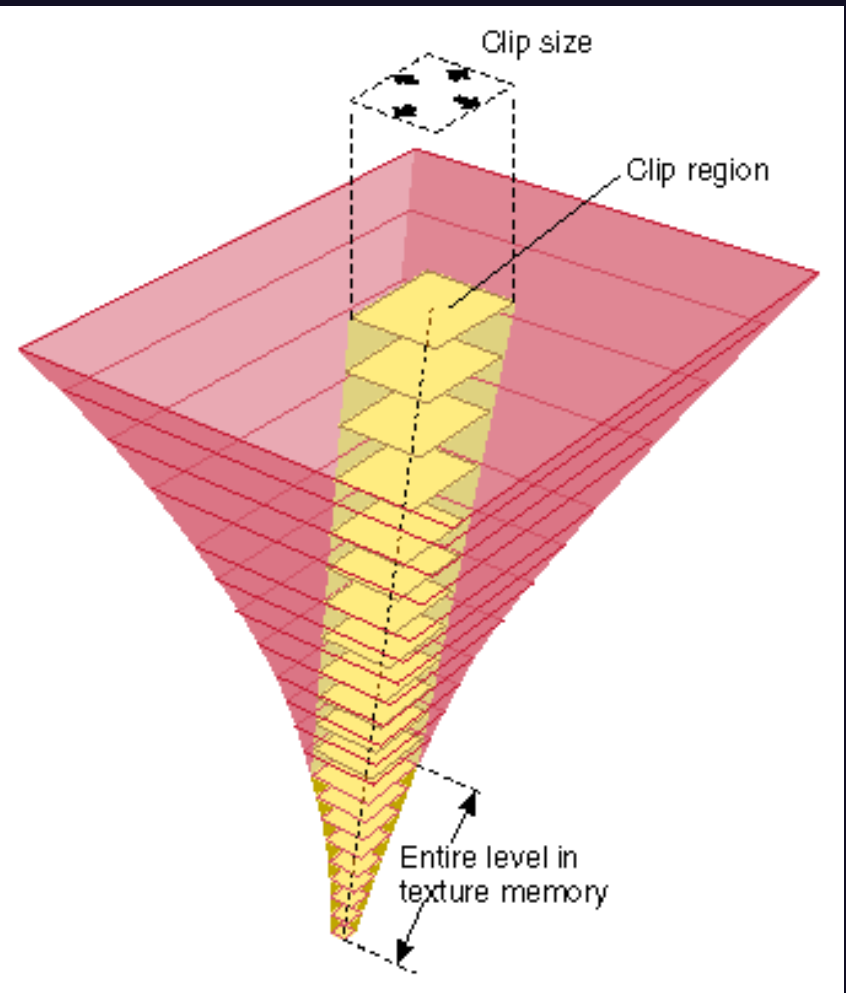
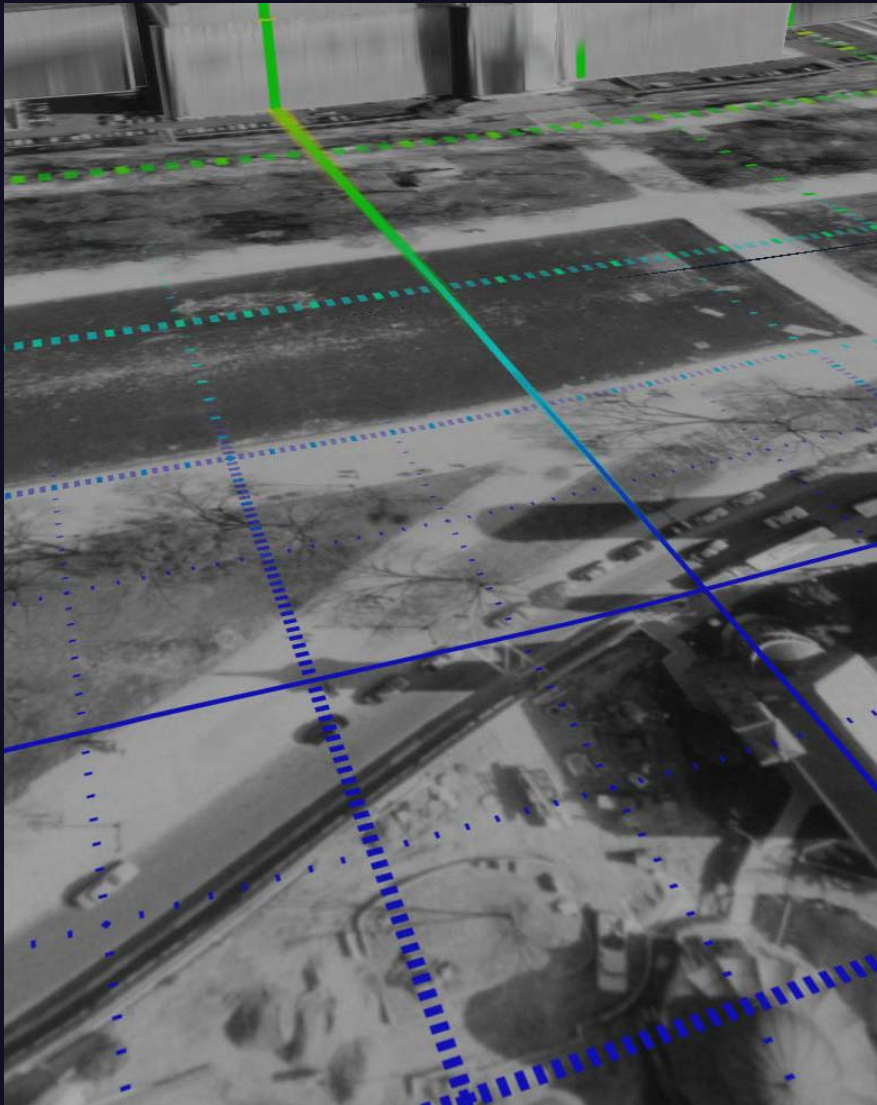
- Expand the concept of sensor projectors to accurately project imagery collected by virtually any type of optical sensor (for instance, pushbroom, whiskbroom, PAN, SPOT, etc). This will increase the breadth of applications for which sensor projectors can be utilized.
- Extend the piecewise frame sensor projector to provide efficient means of storing, moving (internet), and projecting very large quantities of high-resolution imagery from all types of optical sensors.

3D Visualization Approach

- Piecewise frame sensor modeling. Any type of sensor projector will be approximated as a collection of one or more frame sensor projectors.
- Essentially all of the optical sensor types utilized in photogrammetry are represented in the BAE Systems library of sensor models.
- For each sensor type represented, the library has a function that for a given point in the image yields the ray along which light traveled to arrive at that point. This function is known as the imaging locus function.
- Over any sufficiently small section of an optical sensor's image, this function can be approximated as the imaging locus function of a frame sensor with a particular set of parameters.

Approach (continued)

- Develop methods that determine the number of sections into which an image must be divided so that a sufficiently accurate (better than one pixel) frame sensor representation can be determined for each section.
- The simultaneous projection of all the sections of the image, utilizing frame sensor models, will be the equivalent of projecting the entire image using its rigorous sensor model.
- Clipmapping - adapt cliptexture concepts for piecewise frame sensor modeling to run efficiently across computer networks.



Status

- Autoclip
- ProjectorSensorModelGroup::()
- Divided frame model
- ray() - piecewise frame function
- Classified VESPER lab established

Summary

- Foundation is in place for push to completion
- If IKONOS sensor model becomes available unclassified demonstrations will be possible



VESPER for the Battlespace (VFB)

Rapid and Affordable Sensor-to-Visualization



Joseph R. Spann
BAE SYSTEMS – Mission Solutions
San Diego CA (858) 592-5853
joseph.spann@baesystems.com

Photogrammetry Using 3-D Graphics and Projective Textures

Joseph R. Spann and Karen S. Kaufman

BAE SYSTEMS Mission Solutions



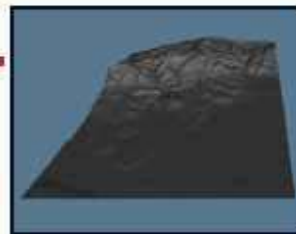
Oblique photograph of
Boulder, Colorado



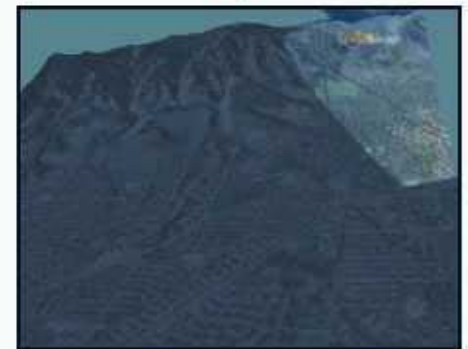
Aerial photograph of
Boulder, Colorado



Sensor models



Terrain/feature model of
Boulder, Colorado



Multiple images simultaneously projected
on terrain and features

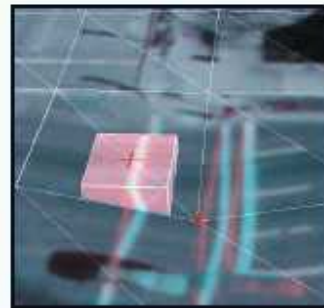
Sensor projector combines rigorous photogrammetric camera models with 3-D virtual reality



Imagery projected on terrain/feature
models, Washington, D.C.



Orthographic view yields true
orthographic projection

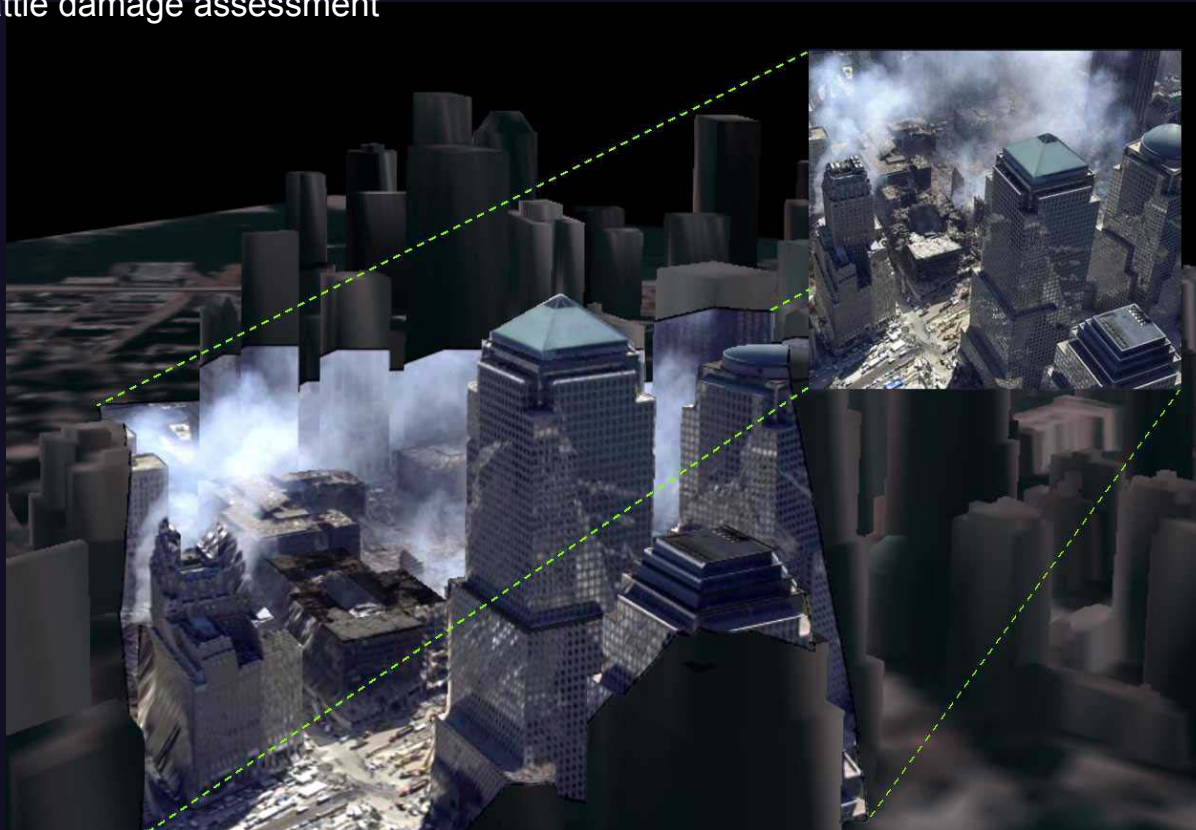


Movable projection surface enables
photogrammetric measurements

BAE SYSTEMS

Military Impact

- The VFB will augment the COP to achieve a rapid and affordable "Sensor-to-Visualization" process for time-critical operations such as:
 - intelligence operations
 - mission planning and rehearsal
 - targeting initiatives
 - operational command and control
 - battle damage assessment



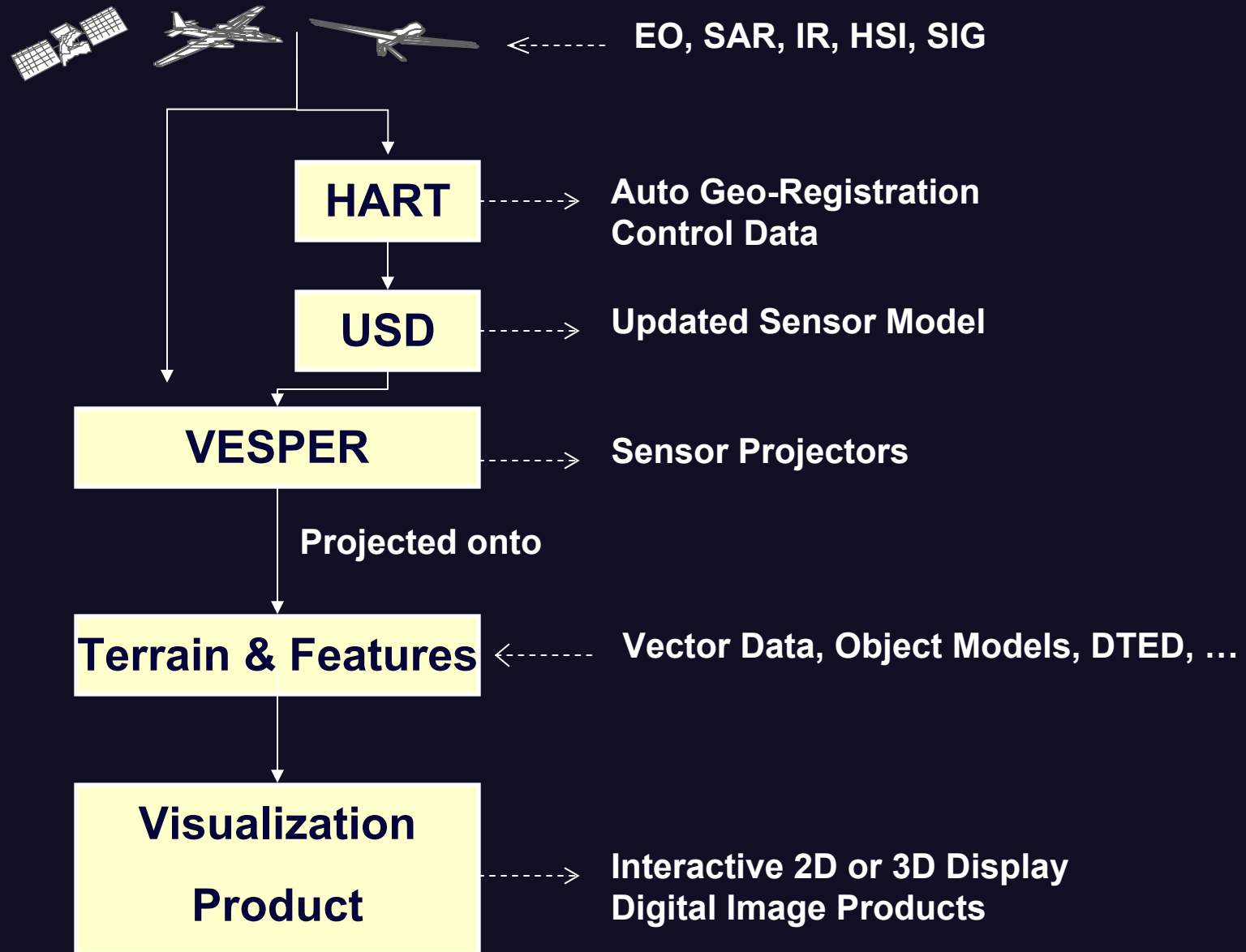
VFB Objectives

- Develop and demonstrate advanced sensor data fusion, 2-D and 3-D visualization capabilities to enhance the Joint Vision 2020 requirement of Dominant Battlespace Awareness.
- Combine sensor visualization techniques (VESPER) with precise geo-registration creating a single data visualization system.
- VESPER + precision geo-registration = end-to-end sensor to visualization
- Enhance the COP with new capabilities for multi-dimensional Dominant Battlespace Visualization.
 - Efficient scalable solution



Provide an integrated "Sensor-to-Visualization" process for time-critical operations such as intelligence operations, mission planning and rehearsal, targeting initiatives, and operational command and control.

VFB Data Flow



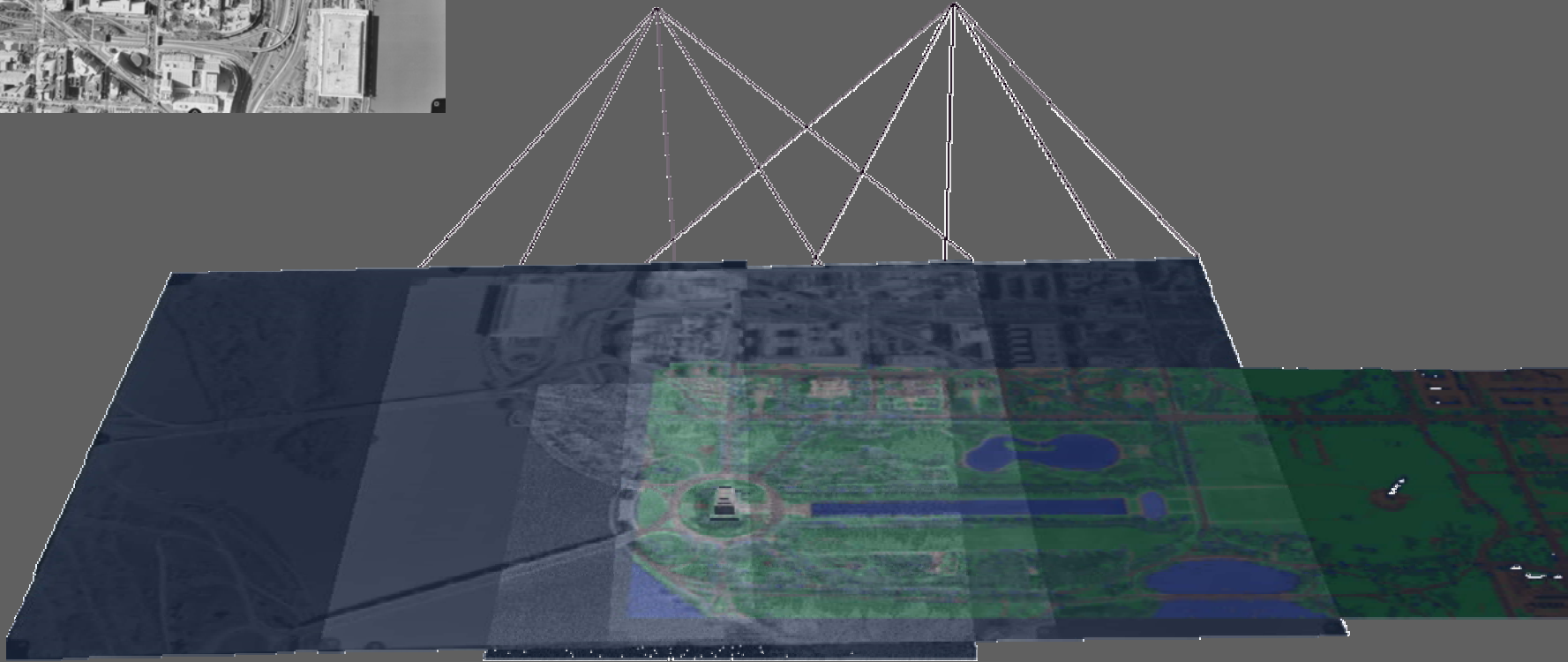
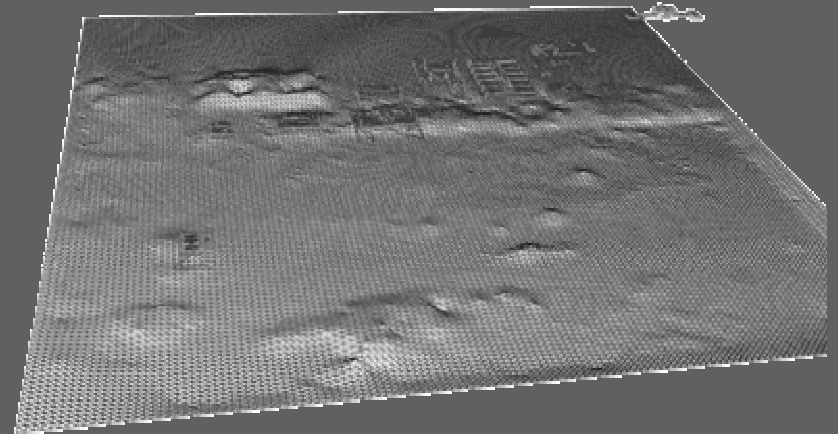
VFB Improves Current Capability

- VFB eliminates the need for time consuming data preparation
 - Foundation data prep currently takes from days to weeks
 - Current scene visualization can take days to produce and only addresses a specific predefined scenario.
 - VFB automates the process for precise geo-registration and sensor modeling.
- VFB allows the real-time integration of all data sources.
 - Current systems often require image rectification and image to image correlation, resulting in a loss of information and precision.
 - Sensor projectors utilize rigorous sensor models with geo-registration allowing for immediate and precise presentation and fusion of any data source.
- VFB delivers ad-hoc source selection and fusion from any perspective.

VESPER: Reverse the data collection process



Sensor Data
projected onto
Terrain and
Feature Data

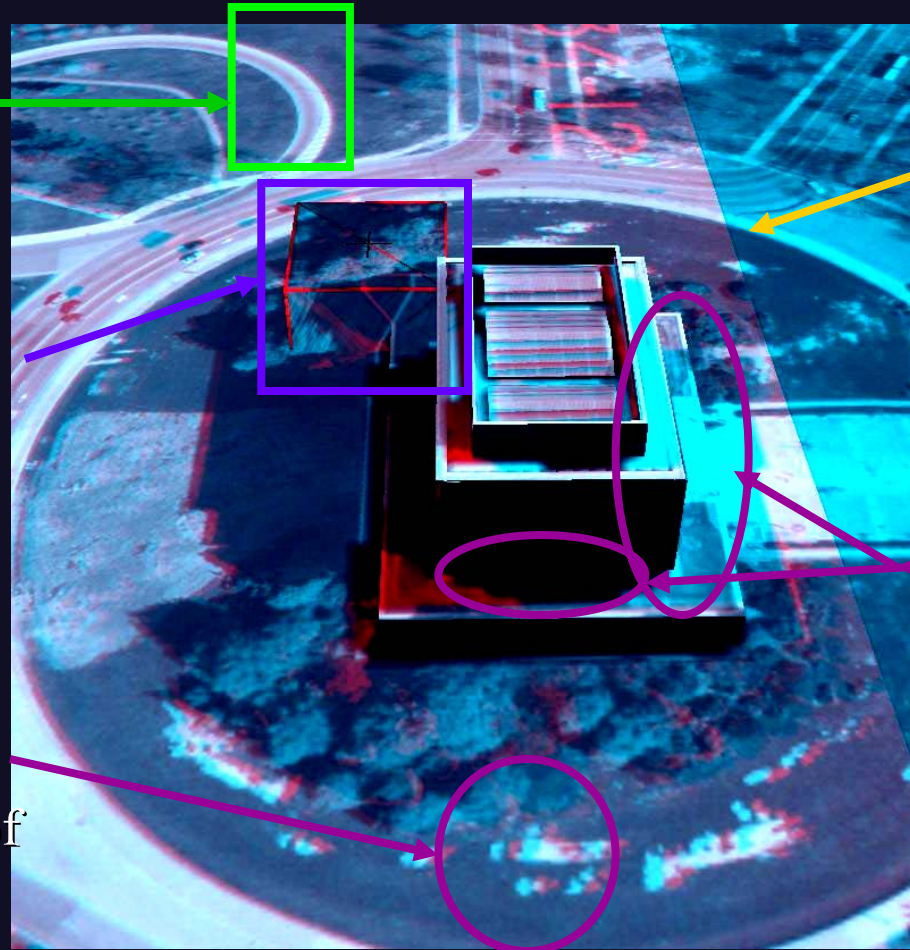


Multi-Image Projection with Color Separation

Grayscale confirms accuracy of surface model

Mensuration using cursor cube

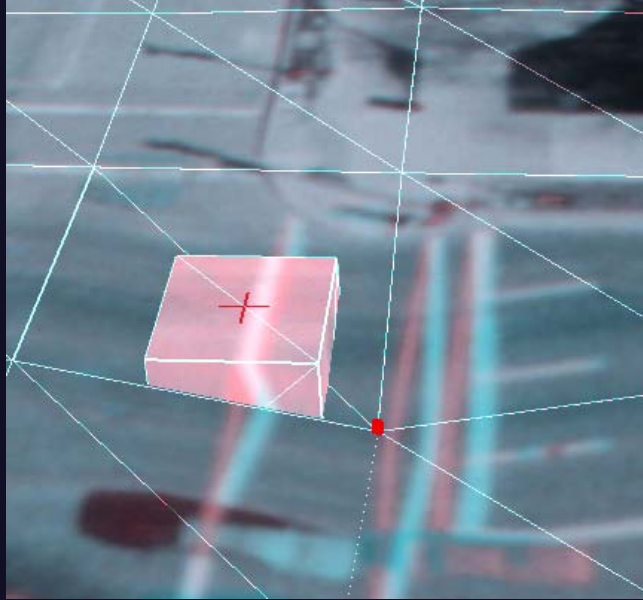
Identification of surface model errors



Edge of image overlap

Occluded
areas
identified

Mensuration and Model Refinement



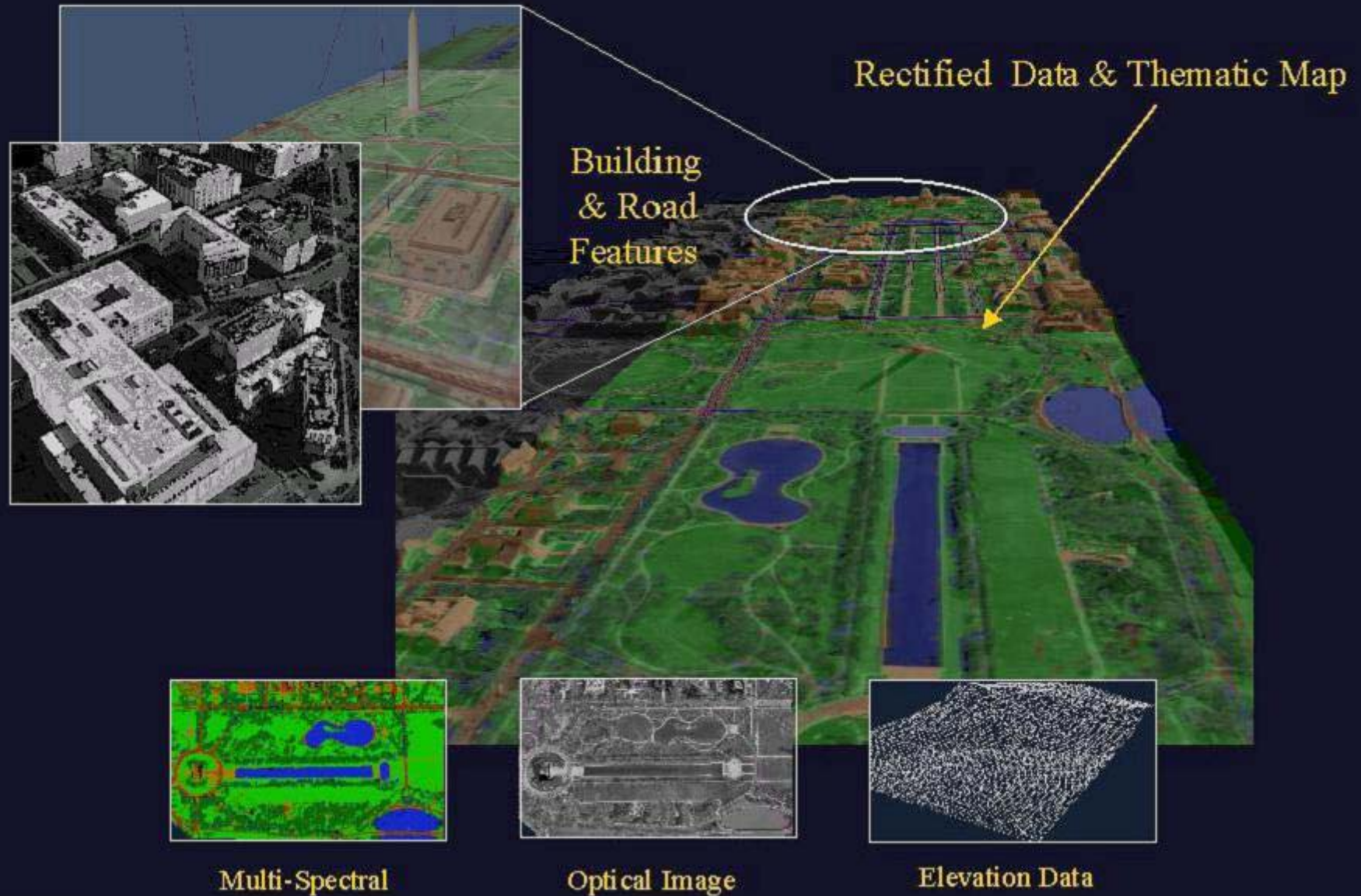
- Refine world model with surface edits until images have a satisfactory alignment

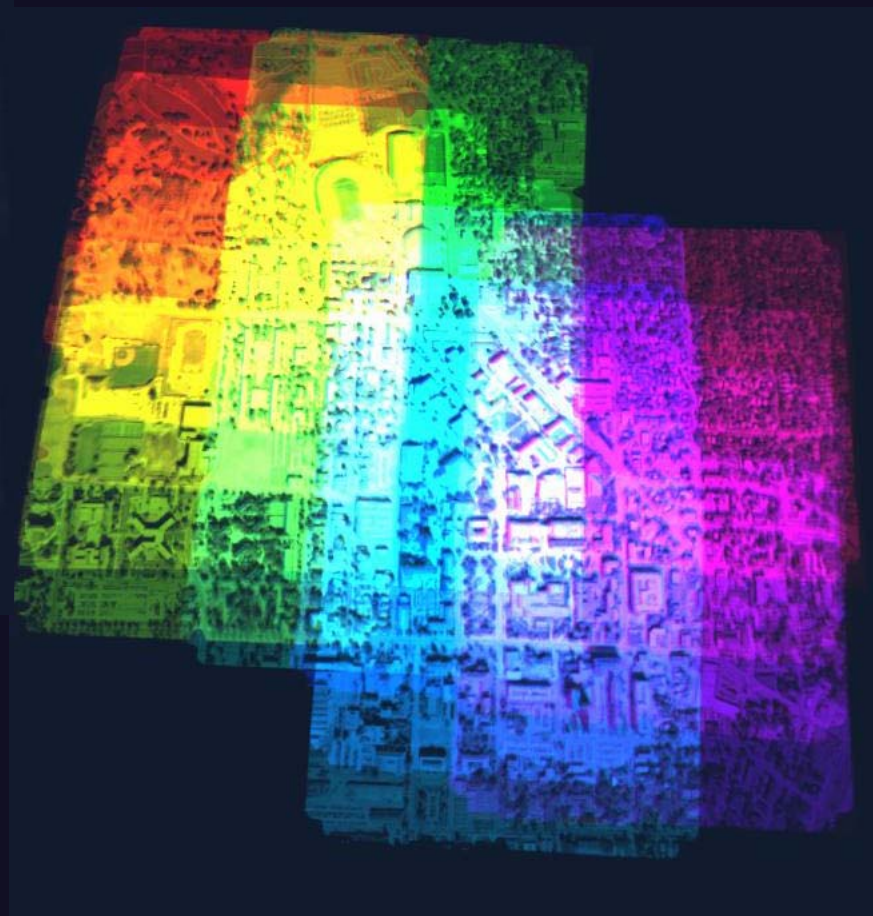
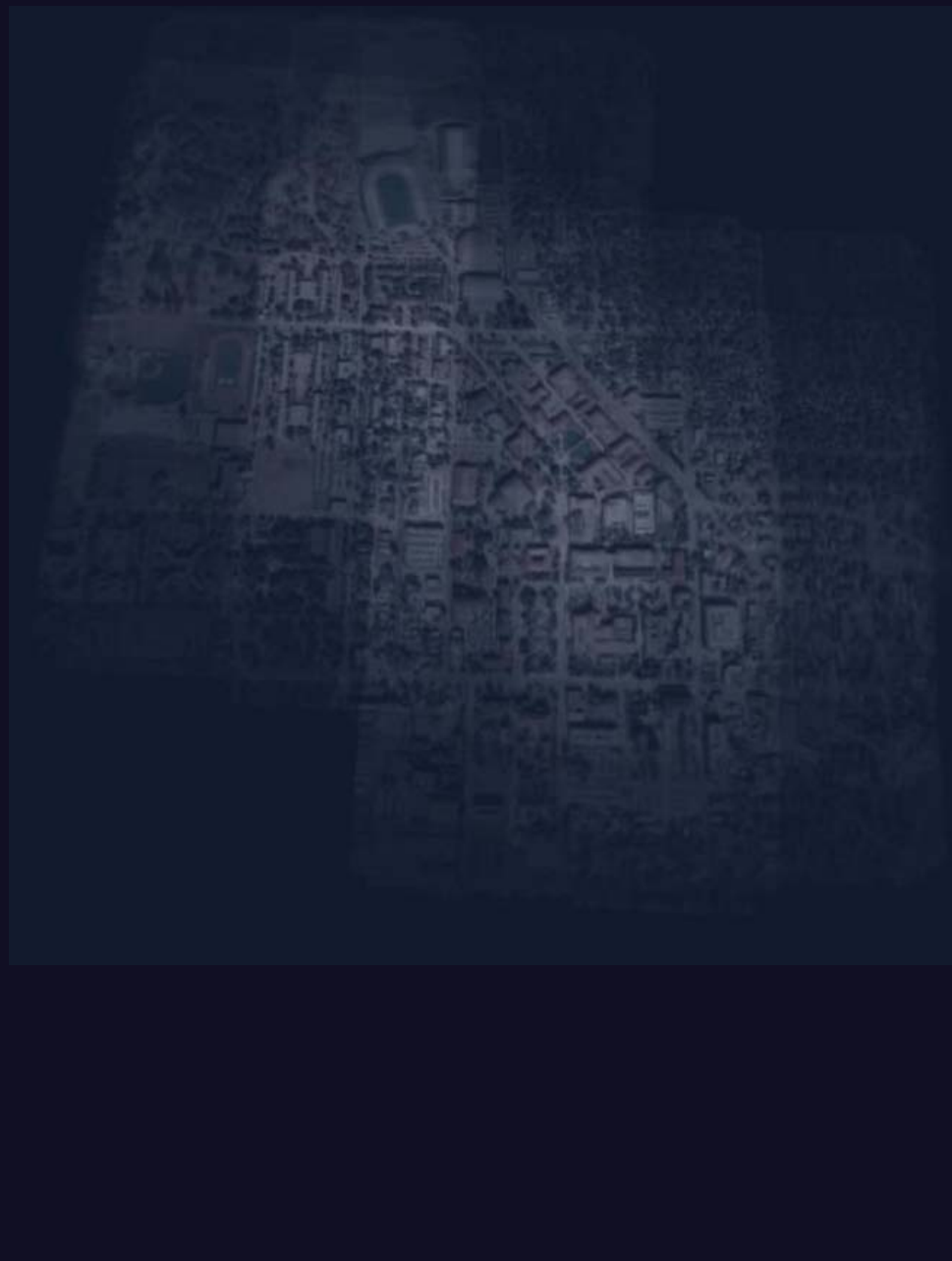


The cursor cube was used to identify a 6.2 foot elevation error in the Digital Terrain Model (DTM).

EXAMPLE 3-D MULTI-IMINT SOURCE FUSION

BAE SYSTEMS





Multi-Image Mensuration via Focus



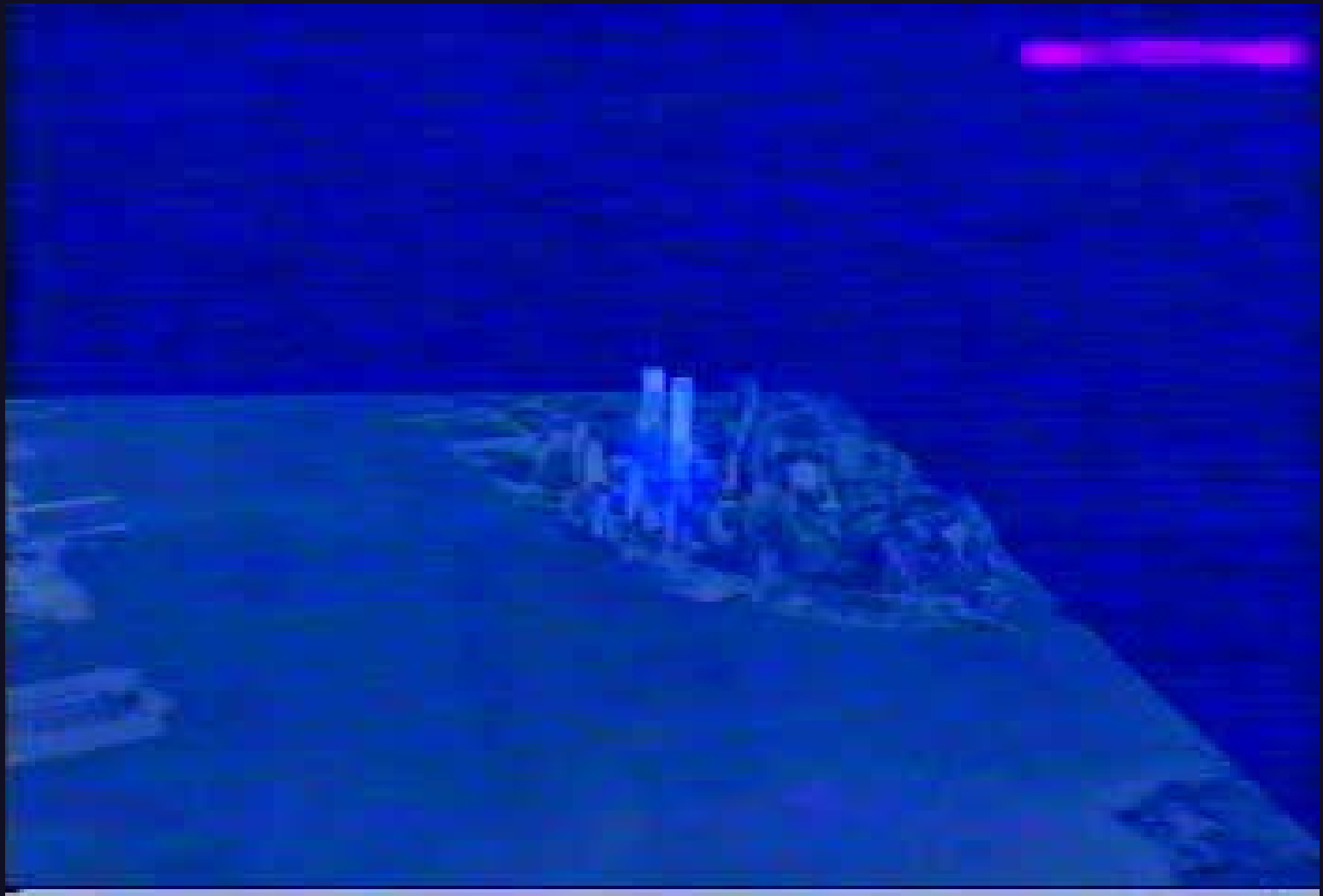
Registered Video Projected to Reference Data



Multi-Int Data Fusion



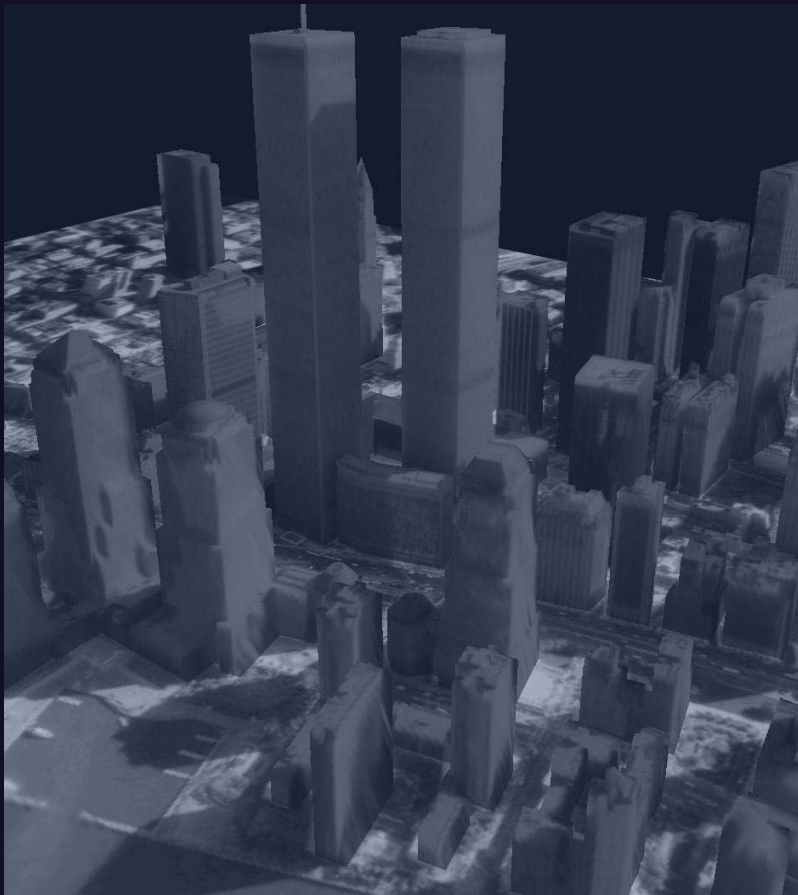
New Imagery Projected to Reference Data



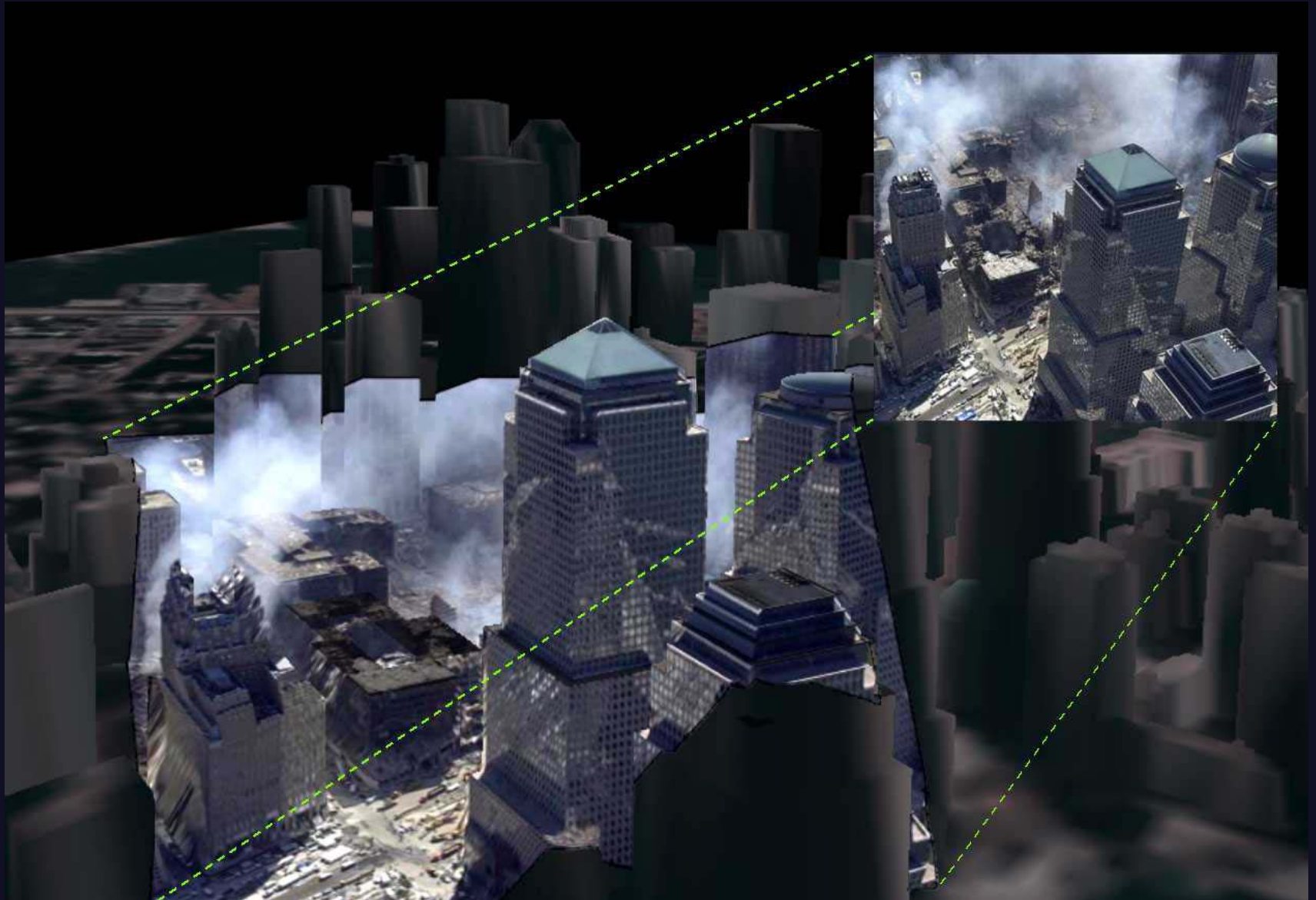
VFB Current Status

- VESPER is the result of 5 years of basic research funded by ARO FY96 - FY01 \$1M
 - The Visualization Environment Supporting Photogrammetry and Exploitation Research (VESPER) test bed was used to produce the examples in this presentation.
 - Capable of presenting 30+ sensor types, tactical to NTM, from multiple simultaneous sources (many gigabytes of data) in 2D or 3D displays on the desktop or over networks.
 - This innovation allows many very large images to be viewed in the context of the underlying three-dimensional terrain and features. This enables new visual methods for multi-image photogrammetry, multi source data fusion and exploitation.
- 3D Visualization funded by ARO FY02 \$86K
 - Further support for NTM data.
 - Immersive VR demonstration
- HART was a MERIT program with FY01 funding \$498K FY00 \$499K
 - Geo-Registration of EO,IR,SAR to within 2 pixels.
 - Fully automates the registration process

New imagery integrated with existing 3D NYC in minutes



CNN Image Resection



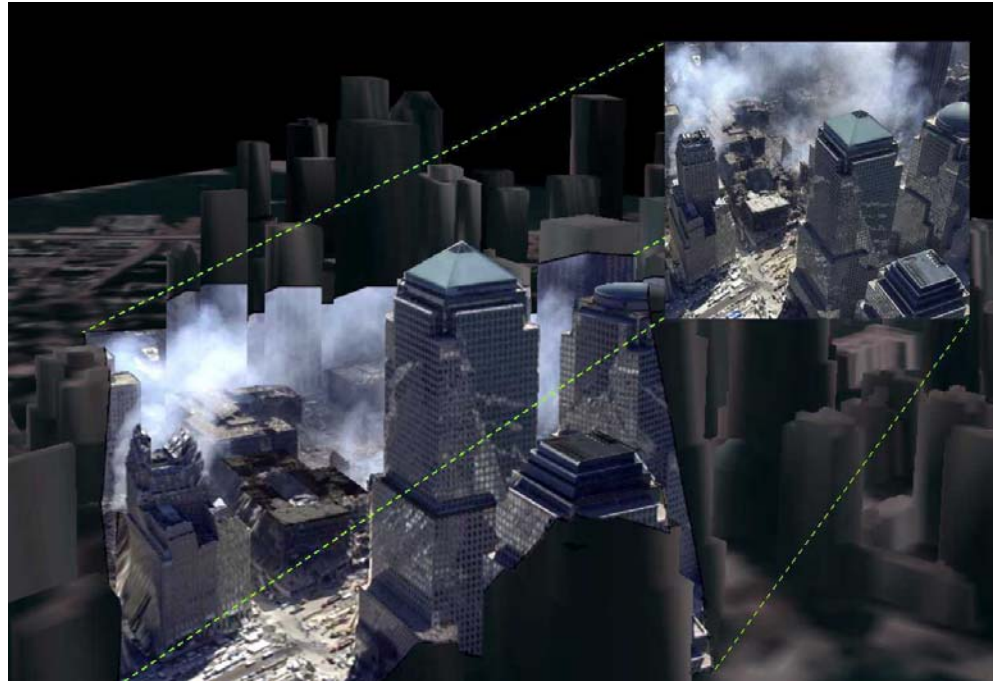
Visualization timelines for NYC

- The feature database was pre-existing from a 1995 ant-terrorism analysis of the WTC. If it hadn't been, USGS DEM data would have been loaded for the terrain, a 5 minute download off the internet, and features would be either retrieved from other sources or modeled in VESPER using our Get3D modeling tool. Usually 2 minutes per building for moderate complexity. SocetSet to OpenFlight conversion took 10 min.
- The black and white aerial frame images over NYC were from the 1995 collection so they were precisely geo-registered previously. Imagery and support data had to be prepared for VESPER. ~5 minutes.
- The new image was off the internet without support data. An interactive resection was performed using 4 - 6 ground-to-image points. A 5 to 10 minute operation using a clumsy lab tool. This manual process can be reduced to 20 seconds with appropriate interfaces.
- The videos are real-time output from VESPER. NTSC is fed to a PC with an MPEG encoder card so time is length of video. Typically 1-3 minutes. Since it's easy I often do 3 or 4 "takes" and pick the best.

Generic Object Placement



Camera resection



Presented by: *Joseph Spann*

Nomination Details

- What is the nomination?

Dr. Taylor, Dr. Krinsky, and MS. Kaufman have developed and implemented camera resection algorithms that allow images from unknown sources to be managed with the same rigor as controlled imagery .

- Why was there a need for this activity?

There is a need to quickly understand image data from uncontrolled sources (CNN, NBC, Al Jazeera ..) in a precise geospatial context.

- How has the nominated activity been developed?

The initial resection algorithm was developed for customers in the intelligence community for processing Predator data and has been implemented on PTW and EPPIC. This work has been generalized for broader applicability and evaluated within the visual context of the Mission Solutions Advanced Programs Visualization activities.

Contribution from Nominees

- Originality/Novelty - how original is the idea / application / approach taken? *This was previously achievable only with controlled intelligence assets. Previous attempts at resection suffered from numerical instability. The strength of the this approach comes from the robustness of its solution.*
- Initiative - how much initiative did the nominee use in bringing the idea to fruition? This approach was initiated by the team to solve a basic image processing problem.
- Level of Effort - how much time and energy was applied by the nominee to develop and implement the innovation or in encouraging others to do so? This innovation was developed through many man months of concerted effort by the highly qualified team.

Business Benefit

- Has the innovation led to improved business performance?

This technology enables the exploitation of many new image sources.

It has been implemented in current products such as JTW and EPPIC and included in several new business proposals.

- What financial performance improvement or time savings have been realized?

It used to take 4 hours and required a skilled photogrammetrist to perform a resection. Now it can be done in under a minute with greater accuracy.

Technology

- **Have any new innovations being generated as a result of this technology?** The VESPER resection tool is based on this innovation. APY-6 risk mitigation applications.
- **Has the innovation resulted in a competitive advantage for the company?** This is a BAE Systems unique solution that is superior to the competition's rubber sheeting approach due to the sensor model rigor that is applied. This technology was included in new business proposals for "Situational Awareness for Field Units" (Homeland Defense) and "C3 Systems Discovery and Invention Program for Visualization" (ONR).
- **How efficient was the application of this innovation in terms of cost, timescales, level of disruption etc.?** Prior to this resection algorithm it usually took 4 hours and required a skilled photogrammetrist to perform a resection. Now it can be done in under a minute with greater accuracy.
- **How does this product or service appeal to the customer?** Data visualization in a common relevant operating picture (CROP) is a first order requirement to achieve Joint Vision 2020. Camera resection allows more data sources to be available for the CROP.

Applicability Across the Company

- What level of company support did the innovation have?

Mission Solutions Advanced Programs provided all the necessary resources and collaboration required to develop this innovation. This included funding under the Advanced Programs Visualization IRAD.

- What plans are there for sharing the innovation locally, within your business unit and across BAE SYSTEMS?

This camera resection work is being applied across business units already.

Impact on Company Values

- Furthering the Company Values - to what degree has the nominees action and behaviors impacted any or all of the Company values (People, Customers, Partners, Technology and Performance)?
- Customer: US ARMY Research Office, NIMA, Intelligence Community
- People: This team effort required imagination and deep understanding of sensor modelling, mathematics and visualization.
- Partnering: All users of geospatial and intelligence imagery can benefit from this innovation.
- Innovation and Technology: This multi disciplinary innovative technology was developed through interaction and sharing of expertise in mathematics, graphics, photogrammetry and sensor modelling. A collective capability unique to Mission Solutions.
- Performance: The resection algorithm allows imagery from unknown sources to be controlled and understood in a precision geospatial context within a few minutes.

Benefits Summary

- Top three benefits resulting from the innovation
 - 1 *This innovation allows images from unknown sources to be managed with the same rigor as controlled imagery . This enables images from CNN or Al Jazeera to be understood in a precision geospatial context.*
 - 2 *The algorithms are fast and easy to use. New images can be introduced in seconds instead of hours.*
 - 3 *These algorithms make use of object oriented design technology, providing our business unit with capabilities applicable not just to the current systems but future systems as well.*

Supporting Information

The BAE SYSTEMS proprietary frame resection algorithm uses a set of at least four ground points and their corresponding image positions to find the position, attitude, and focal length of the camera. The interior orientation coefficients are assumed to be known. Suppose that a set of ground points (x_{Gi}, y_{Gi}, z_{Gi}) and the corresponding image points (l_i, s_i) are known. We assume that the focal plane coordinates (x_{Fi}, y_{Fi}) corresponding to the image coordinates (l_i, s_i) can be found.

The focal plane coordinates can be written explicitly in terms of the ground coordinates, the aperture position (x_S, y_S, z_S) , the effective focal length d , and the rotation matrix R that describes the camera attitude. Note that the focal plane coordinates are a linear rational transformation of the ground coordinates.

Precision Georegistration in seconds

BAE SYSTEMS

